МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

4 | 1984

ISSN 0233-4844

● **Программирование: наука, технология, искусство!** Концепции и точки зрения

● **Диалоговые вычислительные комплексы ДВК «Электроника НЦ-80-20»** — массовый инструмент обработки информации для инженеров, технологов, научных работников, сотрудников учреждений и т. д.

● **Магистрально - модульные системы** различной конфигурации на базе комплекта однокристального микропроцессора серий К1801/К1809 и одноплатных микропроцессорных модулей типа «Электроника 60», «Электроника С5-21М», «Электроника С5-41» легко строить, наращивать, модернизировать

● **Преимущества оптоволоконных локальных сетей** — широкополосность, помехозащищенность, взрыво- и пожаробезопасность реализуются при использовании созданных отечественной промышленностью систем «Электроника МС 4101, МС 8201, МС 8401, МС 8050»

● **Многооконное текстовое взаимодействие с персональной ЭВМ:** поверхность экрана имитирует поверхность стола с набросанными на нем документами

● **Локальная сеть устройств МС УВТ В7/В9** — микросредств управляющей вычислительной техники предназначена для создания распределенных систем с расстояниями 0,5—1,5 км

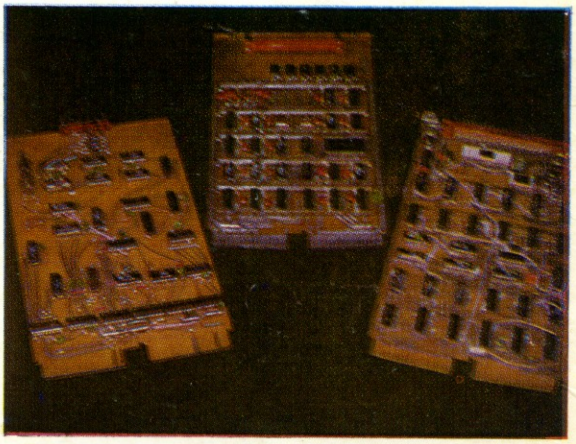


АВТОМАТИЗИРОВАННАЯ СТАНЦИЯ ДЛЯ ДИСТАНЦИОННОГО ИЗМЕРЕНИЯ ПАРАМЕТРОВ АТМОСФЕРЫ РАДИОМЕТРИЧЕСКИМ МЕТОДОМ (ДИПАРМ)

(к ст. А. А. Кочеткова, В. В. Крылова)



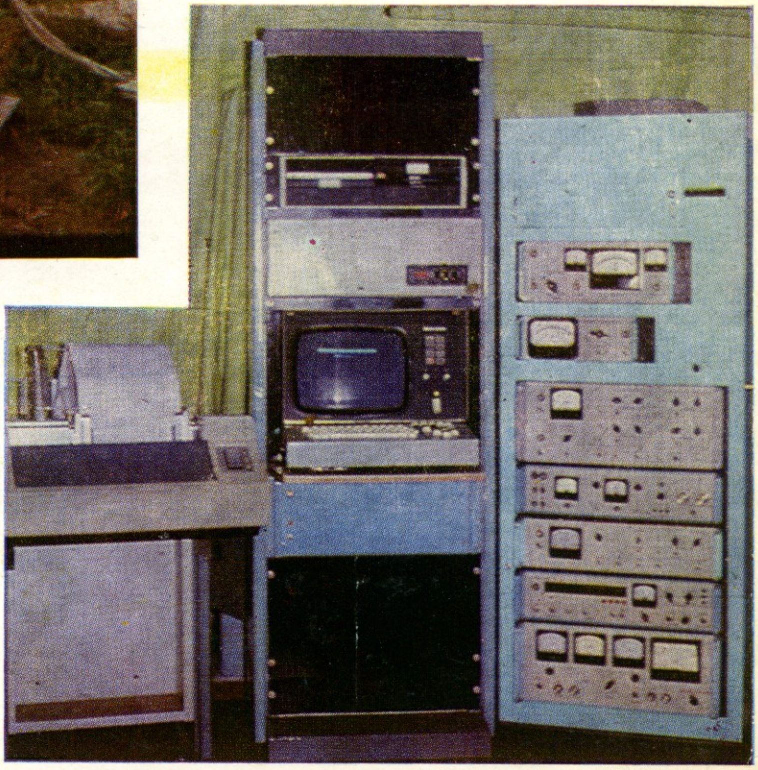
Антенная система станции



Нестандартные устройства сопряжения измерительной аппаратуры станции ДИПАРМ с микроЭВМ

Техническая характеристика станции

Диапазон принимаемого излучения, ГГц	20—100
Погрешность определения параметров атмосферы, %	2
Длительность обработки измерения, мин	10
Интервал между двумя измерениями, мин	15
Объем оперативной памяти, Кбайт	28
Емкость внешней памяти для архивного хранения	465 измерений на одном гибком диске



Метеорологическая станция ДИПАРМ →

Пролетарии всех стран, соединяйтесь!

ОРГАН ГОСУДАРСТВЕННОГО КОМИТЕТА СССР ПО НАУКЕ И ТЕХНИКЕ

Издается с 1984 года

МП МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

ВЫХОДИТ ЧЕТЫРЕ РАЗА В ГОД НАУЧНО-ТЕХНИЧЕСКИЙ И ПРОИЗВОДСТВЕННЫЙ ЖУРНАЛ 4 | 1984 МОСКВА

СОДЕРЖАНИЕ

Адрес редакции: 101820, Москва, проезд Серова, 5, редакция журнала «Микропроцессорные средства и системы». Телефоны 228-18-88; 221-99-26

Сдано в набор 20.11.84 Т-25506
Подписано в печать 25.12.84
Формат 84×108 1/16. Бумага № 1.
Печать высокая Усл. печ. л. 10,08
Уч.-изд. л. 14,9 Тираж 7 000 экз.
Заказ № 1664 Цена 1 руб. 10 коп.

Орган Государственного комитета СССР по науке и технике

Типография Всесоюзного центра информации по оборудованию ГКНТ СССР.
Московская типография № 13 Союзполиграфпрома Госкомиздата СССР

На первой странице обложки: Диалоговый вычислительный комплекс «Электроника НЦ-80-20/3» — ДВК-3

На четвертой странице обложки: Система однокристальных модулей серий К1801/К1809

КОЛОНКА РЕДАКТОРА

Индустрия ЭВМ не отметила еще и полувекового юбилея своего существования, а эффективность функционирования и темпы роста новой отрасли в значительной степени влияют на темпы роста экономики страны в целом. Влияние это становится все более заметным по мере углубления и развития тех радикальных перемен в методах и стиле хозяйствования, которые связаны с намеченным XXVI съездом КПСС и последующими Пленумами ЦК КПСС переходом от экстенсивных направлений роста экономики страны к интенсивным.

Экстенсивные методы хозяйствования основывались, на постоянном расширении масштабов промышленной эксплуатации природных ресурсов, как правило, не возобновляемых. В противоположность этому, принципиально ограниченному направлению развития, интенсивные методы — ориентированы на эксплуатацию постоянно наращиваемых интеллектуальных ресурсов, так называемых, национальных информационных ресурсов. При этом резко возрастает экономическая значимость отрасли, занятой производством станков для обработки информации — ЭВМ. Информационные машины — ЭВМ рассматриваются в промышленно развитых странах как универсальный «двигатель», основной привод самых различных хозяйственных механизмов повышения производительности труда.

Иными словами, исторически новая отрасль народного хозяйства — индустрия ЭВМ, обеспечивает **производство средств производства** *для промышленной эксплуатации национальных информационных ресурсов и создает, таким образом, необходимые технические предпосылки для интенсификации практически всех отраслей народного хозяйства. Согласно данным, которые приводит заместитель председателя ГКНТ СССР А. К. Романов, «микропроцессоры в ближайшие 8—10 лет найдут применение более, чем в стах тысячах различных видов устройств, приборов и установок промышленного и бытового назначения» («МП», № 1, 1984 г.).*

Одна из главных в настоящее время трудностей на пути массовой компьютеризации народного хозяйства — необходимость в короткий срок вооружить многомиллионный инженерный корпус страны первыми элементами культуры компьютерного мышления и определенным минимумом конкретных технических знаний в области применения микроЭВМ и микропроцессоров. С целью оказать практическую помощь отраслям и ведомствам в решении этой масштабной задачи ГКНТ СССР начал в 1984 году издание массового иллюстрированного научно-технического и производственного журнала, посвященного проблемам разработки и внедрения в широкую инженерную практику технических средств и систем на базе микропроцессоров.

Из простого сопоставления четырех вышедших в этом году номеров журнала читатели могут видеть, что процесс формирования художественного и тематического облика нового журнала далеко не завершился. Идет напряженный поиск, вопросов все еще много больше, чем готовых решений и редакция постоянно ищет поддержки и помощи у читательского актива.

Сегодня нам хотелось бы поэтому еще раз привести здесь заключительный фрагмент из выступления А. П. Ершова, которым он год назад открывал первый номер «МП»: Залогом эффективности и полезности массового журнала является его неразрывная связь с читателем и учет его пожеланий. В период становления журнала редсовет, редколлегия и редакция особенно заинтересованы в обратной связи. Мы будем признательны каждому, кто пожелает высказать свою оценку публикуемых материалов и рекомендации в отношении журнала «Микропроцессорные средства и системы».

Г. Р. Громов

УДК 681.32

Б. М. Малашевич, В. Л. Дшхунян, Ю. И. Борщенко, О. А. Никольский, В. С. Хорин

МАГИСТРАЛЬНО-МОДУЛЬНЫЕ МИКРОПРОЦЕССОРНЫЕ СИСТЕМЫ

Микропроцессорные средства и системы (МСС) обладают уникальной совокупностью технико-экономических характеристик, сочетая высокую вычислительную и управляющую мощность с малыми габаритами, стоимостью и энергопотреблением. Интегральная технология производства основных устройств МСС обеспечивает возможность их массового выпуска и широкого применения — от детских игрушек до сложных промышленных, транспортных и других систем, что, в свою очередь, определяет основные требования, которым должны удовлетворять МСС и принципы их построения.

Требования к микропроцессорным средствам и системам

Микропроцессорные системы можно условно разделить на две основные группы: **управляющие и вычислительные.** Управляющие МСС должны встраиваться непосредственно в аппаратуру управляемого объекта, вычислительные — свободно размещаться на рабочем столе инженера, ученого, служащего. Следовательно, все или большинство устройств МСС должны быть конструктивно скомпонованы в одном моноблоке, т. е. быть встраиваемыми. Отсюда вытекает несколько требований к микропроцессорным средствам: для них не требуются (или необязательны) индивидуальные источники питания, органы управления и индикации, конструктивно-декоративная завершенность (кожух, лицевая панель и т. п.).

Необыкновенно широкая область применения МСС определяет соответственно высокую потребность в них, т. е. обусловливает необходимость массового производства. Отсюда требования к технологичности их изготовления и необходимость управления номенклатурой с целью ее минимизации. Потребители МСС — в основном не профессионалы в области микроэлектроники и вычислительной техники, поэтому важно, чтобы МСС были не только просты в эксплуатации, но и допускали применение упрощенных методов конструирования.

Область применения МСС расширяется постепенно, с нарастающим темпом, в результате чего появляется необходимость развивать действующие микропроцессорные системы, объединять несколько систем в более сложные, адаптировать и модернизировать их под изменившиеся задачи и т. п. Широкое применение МСС серьезно обостряет задачу их гарантийного и послегарантийного обслуживания и ремонта, требует высокого уровня надежности и качества МСС и ограничения их номенклатуры.

Таким образом, МСС должны иметь минимизированную номенклатуру, выпускаться массовыми тиражами во встраиваемых и автономных исполнениях, соответствовать требованиям совместимости (в первую очередь по интерфейсам, конструкции и программному обеспечению), обеспечивать простоту построения, наращивания, модернизации и реконфигурации прикладных систем, обладать высокой надежностью. Удовлетворение всех этих требований возможно только на основе комплексной унификации и магистрально-модульного принципа построения.

Магистрально-модульный принцип построения радиоэлектронной аппаратуры

Этот принцип впервые получил широкое распространение в разработке фирменных и международных систем унифицированных модулей NIM, КАМАК, ВЕКТОР, используемых при создании управляющих систем в области ядерной физики. Эти системы модулей были реализованы на основе дискретных изделий электронной техники, выполняли относительно простые функции, а построенные на их основе управляющие системы были сложны, дороги, избыточны. Но они имели два неоценимых преимущества:

совместимость, т. е. в рамках каждой системы модулей были строго стандартизированы конструкции и интерфейсы, что обеспечило потребителю возможность комплексировать прикладные системы из модулей, приобретенных у различных изготовителей;

гибкость, обеспечивающую возможность быстрого построения, наращивания, модернизации или реконфигурации управляющих систем.

Именно поэтому магистрально-модульные системы получили широкое применение в первую очередь при создании уникальных систем управления ядерным реактором и часто реконфигурируемых систем автоматизации научного эксперимента.

Сущность магистрально-модульного принципа построения заключается в том, что прикладная система компонуется из стандартных и унифицированных функционально и конструктивно законченных совместимых модулей. Конструкция модулей обеспечивает возможность их механического объединения в соответствующем корпусе. Обычно модули вдвигаются по направляющим в корпус с одновременным сочленением соединителя на задней кромке модуля с ответной частью на задней стенке объединяющего корпуса. Электрическое и информационное объединение модулей осуществляется через соединитель по магистральному системному интерфейсу.

Микропроцессоры вдохнули новую жизнь в магистрально-модульные системы, распространив принципы их построения на вычислительную технику, промышленное оборудование и другие виды управляющих систем. Функциональная емкость модулей значительно по-

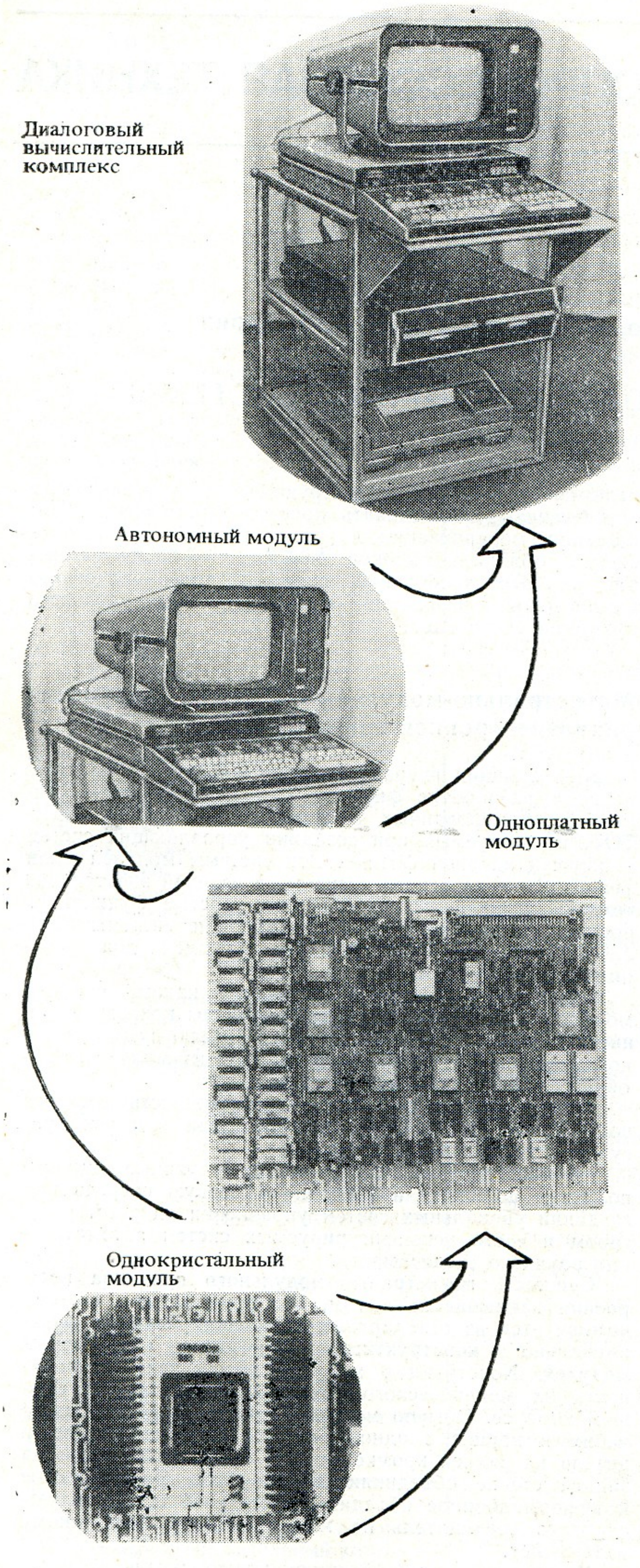


Рис. 1. Иерархия микропроцессорных модулей

высилась, появилась возможность построения функционально законченных модулей в виде микросхемы, одноплатного изделия и т. п. Системы унифицированных модулей приобрели многоуровневую иерархическую структуру (рис. 1), сохраняя свои основные черты на каждом уровне. В настоящее время все большее распространение получают однокристальные модули, унифицированные фрагменты печатных плат [1], вдвижные и автономные модули.

Все виды модулей должны удовлетворять требованиям совместимости, в первую очередь по интерфейсам конструкции и программному обеспечению.

Интерфейсы. Особое значение имеет системный обычно магистральный интерфейс, по которому осуществляется обмен информацией между всеми (или основными) модулями микропроцессорной системы. Выбор этого интерфейса определяется архитектурой микропроцессора. Отечественная микроэлектроника предлагает потребителю два типа интерфейсов:

с разделенными шинами для передачи адреса и данных — **И-41** (ОСТ 25 969-82). Для построения систем на основе этого интерфейса выпускаются 8-разрядный (серии КР580) и 16-разрядный (серии КР1810) микропроцессорные комплекты;

с совмещенными шинами для передачи адреса и данных — **МПИ** (ОСТ 11.305.903-80). обеспеченный микропроцессорными комплектами БИС серий К1801/К1809, К1811, К581, К588 и другими.

Системный интерфейс может иметь внутриплатную, межплатную и межблочную реализации. В будущем интерфейсы И-41 и МПИ должны быть заменены единым машиннонезависимым, инвариантным к разрядности данных и адреса интерфейсом [2, 3].

Кроме системного интерфейса, представляющего в МСС *первый уровень,* применяются интерфейсы других уровней, составляющие в совокупности иерархическую систему (рис. 2).

Ко второму уровню относятся малые интерфейсы периферийных устройств (ПУ) и устройств связи с объектами (УСО). Малые интерфейсы используются только в тех случаях, когда ПУ и УСО не имеют встроенного системного интерфейса и не могут подключаться непосредственно к системной магистрали. Наиболее широкое применение нашли радиальные интерфейсы СМ ЭВМ: последовательный ИРПС (НМ МПК по ВТ 10-78)* и параллельный ИРПР (НМ МПК по ВТ 29-80). При помощи этих интерфейсов подключаются практически все периферийные устройства (дисплеи, печатающие устройства, клавиатуры, пульты, графопостроители и т. п.). за исключением внешних запоминающих устройств (ВЗУ) предъявляющих более высокие требования к пропускной способности интерфейса. В этих случаях применяются специальные интерфейсы, унифицированные по видам ВЗУ, например, интерфейс накопителя на гибких магнитных дисках (НМ МПК по ВТ 12-78). В качестве интерфейса УСО может быть использована магистраль КАМАК, однако на практике модули УСО обычно подключаются непосредственно к системной магистрали И-41 или МПИ. Сопряжение

---

* Нормативный материал (НМ) Межправительственной комиссии по сотрудничеству в области вычислительной техники (МПК по ВТИ: НМ МПК по ВТ 10-78. СМ ЭВМ. Интерфейс для радиального подключения устройств с последовательной передачей информации ИРПС.

НМ МПК по ВТ 12-78. СМ ЭВМ. Накопители на гибких магнитных дисках двухдисковые. Интерфейс. Структура и состав. Требования к функциональным характеристикам.

НМ МПК по ВТ 29-80. СМ ЭВМ. Интерфейс для подключения устройств с параллельной передачей информации (ИРПР).

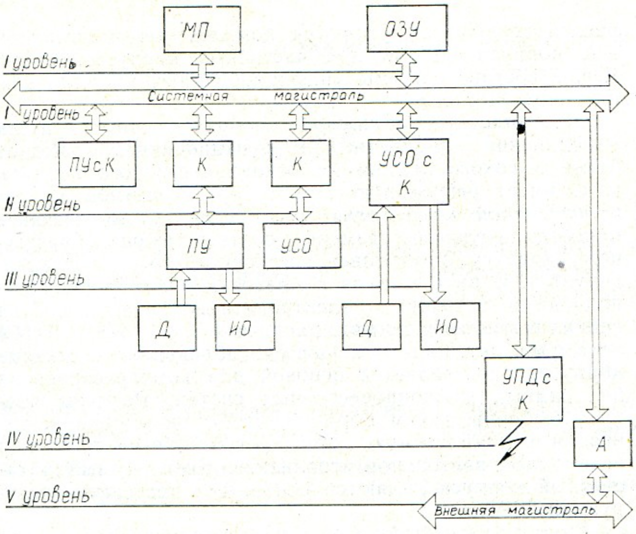


Рис. 2. Система интерфейсов МСС

малого интерфейса с системной магистралью осуществляется при помощи контроллера (К). Малые интерфейсы используются, как правило, при подключении автономных ПУ и УСО к системе.

К третьему уровню относятся интерфейсы датчиков (Д) и исполнительных органов (ИО), т. е. интерфейсы соединения их с УСО. Отсутствие жесткой унификации этих интерфейсов приводит к неоправданному росту номенклатуры УСО, датчиков и исполнительных органов.

К четвертому уровню относятся интерфейсы устройств передачи данных (УПД), т. е. интерфейсы телеграфных, телефонных, высокочастотных, оптоволоконных и других каналов для передачи данных на большие расстояния, в том числе интерфейсы, применяемые для построения локальных, региональных и других типов сетей ЭВМ.

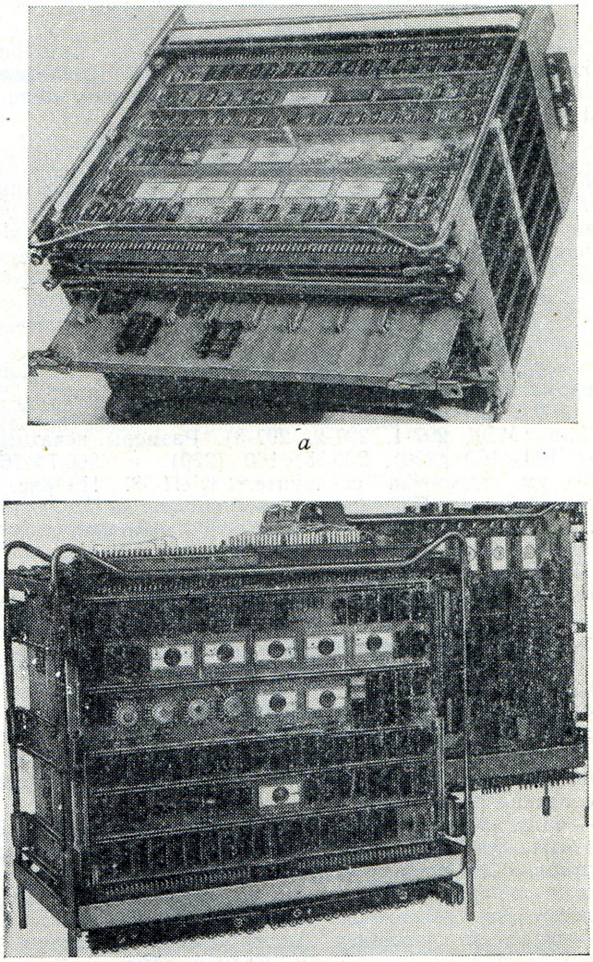
К пятому уровню относятся внешние, относительно микропроцессорной системы, интерфейсы, например, системный интерфейс старшей ЭВМ в многомашинной системе, неунифицированный интерфейс управляемого объекта и т. п. Соединение внешнего интерфейса с системным осуществляется при помощи специального адаптера интерфейсов (А).

Конструкция. Распространение микропроцессоров привело к дальнейшему развитию старых и появлению новых конструктивных исполнений модулей, в первую очередь однокристальных и фрагментов печатных плат. **Однокристальные модули** выполняются в стандартных корпусах интегральных микросхем. В народном хозяйстве получили широкое применение корпуса типа ДИП и КВИП с вертикальными выводами, монтируемыми в отверстия печатной платы.

Фрагментно-модульное проектирование впервые было применено в МСС при разработке одноплатных микроконтроллеров «Электроника С5-41» [1]. Сущность этого метода заключается в том, что зона поверхности печатной платы, предназначенная для размещения электрорадиоэлементов, разделяется на участки стандартных размеров и конфигурации (допускаются размеры, кратные минимальному). На одном из краев такого участка выделяются определенные координатные точки для вывода каждой линии внутриплатной реализации системного интерфейса. В пределах площади стандартного участка производится проектирование топологии функционального модуля с встроенным системным интерфейсом. Таким образом, создается библиотека унифицированных **стандартных фрагментов-модулей,** обладающих интерфейсной, конструктивно-топологической и программной совместимостью. Создание одноплатных МСС производится простой аппликацией унифицированных фрагментов либо автоматически, либо вручную. Метод обеспечивает возможность быстрого, с минимальными затратами, создания заказных одноплатных изделий требуемой конфигурации и изготовления малосерийных изделий на основе массового производства.

Особое значение имеет выбор конструкции **вдвижных модулей,** так как она непосредственно связана с конструкцией управляемого объекта. Размер печатной платы определяет, в основном, структуру, размеры и способы компоновки несущих конструкций радиоэлектронной аппаратуры, а хорошо проработанные несущие конструкции превращаются в конструкционную систему. Структура типовой конструкционной системы содержит три основных уровня: частичный, комплектный и комплексный корпуса.

Частичный корпус является основой для построения одноплатных, многоплатных и произвольной компоновки вдвижных (часто их называют врубными) модулей. Одноплатный модуль представляет собой печатную плату с установленными электрорадиоэлементами, соединителями (системным и, при необходимости, внешними) и рычагами сочленения-расчленения или лицевой панелью. В народном хозяйстве широкое приме-



а

б

Рис. 3. Модули типа «Электроника С5-21М» *(а)* в корпусе, совместимом с микроЭВМ «Электроника 60» *(б)*

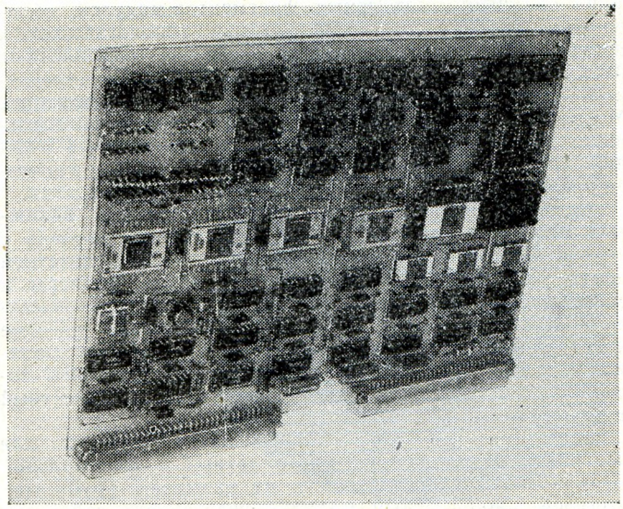


Рис. 4. МикроЭВМ на Европлате 6U

нение получили одноплатные микропроцессорные модули типа:

«Электроника 60» — печатные платы размером 135×240 мм (одинарная) и 280×240 мм (двойная), системные соединители прямого контактирования РППМ-16-72, РППМ-16-288;

«Электроника С5-21М» [4] — печатные платы 7 U по ГОСТ 26.204-83, размером 277,75×220 мм, системный соединитель косвенного контактирования ГРПМ1-61. Модули «Электроника С5-21М» и «Электроника 60» близки по размерам и легко компонуются в совместимые между собой специальные каркасы (рис. 3, а, б);

«Электроника С5-41» [1] — печатная плата 4 U по ГОСТ 26.204-83, размером 144,45×220 мм, системный соединитель СНП-34;

модули международной конструкции на основе «Евромеханики», т. е. на основе печатных плат 3 U, 6 U и 9 U по ГОСТ 26.204-83 (СТ СЭВ 3266-81, Публикации МЭК 297-1, 297-2, 297-3). Размеры печатных плат 100×160 (220), 233,35×160 (220) и 366,7×160 (220) мм, системный соединитель СНП-59. На рис. 4 показана микроЭВМ, построенная на основе печатной платы 6 U (233,35×160 мм) и микропроцессора К1801ВМ1 [5].

Из всех перечисленных конструкций модулей МСС наиболее перспективной является Евромеханика, обладающая совокупностью технических, экономических и эксплуатационных преимуществ:

— обеспечивает компоновочную гибкость конструкции, возможность гармоничного сочетания в одном изделии трех (а с незначительным усложнением шести) размеров печатных плат. Евромеханика — единственная конструкционная система, размерные цепочки которой сбалансированы во всей конструктивной иерархии от соединителя до стойки;

— обладает широким спектром возможных площадей печатных плат и размеров частичных и комплектных корпусов, обеспечивает условия для оптимальной компоновки аппаратуры, снижения ее избыточности, материалоемкости, повышения плотности компоновки;

— предельно упрощает модернизацию прикладных систем: морально устаревшие модули заменяются новыми, с меньшими размерами, освобождается место для установки модулей наращивания системы.

Евромеханика получила широкое применение в различных областях народного хозяйства. В ряде отраслей серийно выпускаются комплектные и комплексные корпуса, детали для частичных корпусов, например, в составе системы унифицированных несущих конструкций «Рябина».

Многоплатный вдвижной модуль строится путем объединения одноплатных в функциональный модуль более высокого уровня на основе частичного корпуса, т. е. имеет регулярную структуру. *Вдвижной модуль произвольной компоновки* имеет стандартные внешние присоединительные размеры, соответствующие частичному корпусу; внутреннее построение его не регламентируется. В виде модуля произвольной компоновки выполняются источники электропитания, накопители на гибких магнитных дисках, дисплеи и т. п.

Комплектный корпус объединяет частичные корпуса, т. е. является основой для комплексирования прикладных микропроцессорных систем. Размеры корпуса выбраны таким образом, чтобы он удобно встраивался в стандартную стойку, тумбу, стол или имел автономное настольное исполнение. Комплектный и частичный корпуса являются основными несущими конструкциями МСС.

Комплексный корпус предназначен для объединения нескольких комплектных и имеет исполнения в виде стойки, тумбы, стола и т. п. С позиции МСС он рассматривается только в качестве несущей конструкции управляемого объекта, в которую должны встраиваться МСС, т. е. быть с ним совместимыми.

В одном из последующих номеров журнала «Микропроцессорные средства и системы» предполагается опубликовать статью, посвященную конструкционной системе, построенной на основе Евромеханики.

Программное обеспечение. Программная совместимость МСС обеспечивается на основе программного обеспечения СМ ЭВМ, содержащего две линии:

— семейства СМ-3/СМ-4 и их развитие. С ними программно совместимы микроЭВМ «Электроника 60, 60М, 60-1, 60-1М, НЦ-80-01Д, НЦ-80-01ДМ, С5-41», микропроцессорные комплекты БИС серий К581, К588, К1801/К1809, КР1811;

— семейство СМ 1800 и его развитие. С ними совместимы микроЭВМ «Электроника К1-10, К1-20, К1-25», МС УВТ В7/В9 и т. д., построенные на основе микропроцессорных комплектов серий КР580 и КР1810.

Для создания программного обеспечения также используется модульный принцип построения, который планируется подробно описать в одном из последующих номеров журнала.

Ниже кратко рассмотрены системы однокристальных и вдвижных модулей.

Магистрально-модульный комплект однокристального микропроцессора серий К1801/К1809

Комплект состоит из функционально-законченных модулей, выполненных в виде БИС. Часть БИС являются многофункциональными, настраиваются на выполнение определенных функций либо программно, либо коммутацией внешних выводов на печатной плате. Совместное применение некоторых БИС позволяет получить новые функциональные модули.

БИС комплекта выполнены на основе N-МОП структур в корпусах типа ДИП и КВИП с числом выводов от 24 до 48 и предназначены для применения в следующих условиях:

напряжение питания, В	+5±5%
емкость нагрузки, пФ	не более 100
температура окружающей среды, °С	от −10 до +70
относительная влажность воздуха, %	98, при температуре окружающей среды +25°С

Таблица 1

Состав комплекта серий К1801/К1809 и краткие характеристики БИС

Обозначение и название БИС	Назначение и характеристика	Число выводов корпуса; №№ выводов питания (+5 и 0 В)
К1801ВМ1 — однокристальный 16-разрядный микропроцессор	Система команд идентична микроЭВМ «Электроника 60», объем адресуемой памяти 64 Кбайт, быстродействие до 500 тыс. операций/с	42; (42 и 21)
КМ1801ВМ2 — однокристальный 16-разрядный микропроцессор	Система команд идентична микроЭВМ «Электроника 60М», объем адресуемой памяти 64 Кбайт, быстродействие до 1 млн. операций/с	40; (40 и 20)
К1809РУ1 — статическое оперативное запоминающее устройство	Емкость 1024×16 бит, время выборки не более 200 нс, цикл обращения не более 400 нс	24; (24 и 12)
К1801РЕ1 — постоянное запоминающее устройство	Универсальное ПЗУ емкостью 4К×16 бит; информация записывается в производственном цикле БИС; время выборки не более 400 нс	24; (24 и 12)
К1809РЕ1 — постоянное запоминающее устройство	То же, время выборки не более 200 нс	24; (24 и 12)
К573РФ3 — репрограммируемое ПЗУ с ультрафиолетовым стиранием	То же, но с возможностью записи и перезаписи информации пользователем; время выборки не более 400 нс	24; (24 и 12)
К1801ВП1-030, К1801ВП1-013 — контроллеры ОЗУ динамического типа	Обеспечивают построение псевдостатического ОЗУ на БИС ОЗУ динамического типа К565РУ3, К565РУ6; используются с микропроцессором К1801ВМ1 и КМ1801ВМ2 соответственно	42; (42 и 21)
К1801ВМ-033 — многофункциональное устройство параллельного интерфейса	Используется для интерфейса НГМД, а в сочетании с К1801ВП1-034 для интерфейса байтового и параллельного обмена ИРПР [2]	42; (42 и 21)
К1801ВП1-034, то же	Используется для мультиплексирования двух байтов с возможностью инверсии, в качестве двухбайтового буферного регистра, компаратора байтов и выработки вектора прерывания	42; (42 и 21)
К1801ВП1-035 — многофункциональное устройство последовательного обмена	Используется в качестве асинхронного приемопередатчика с преобразованием параллельного кода в последовательный и наоборот и обеспечивает работу интерфейса ИРПС [2] со скоростями от 50 до 19200 бод	42; (42 и 21)
К1809ВВ1 — программируемый ввод-вывод информации	Два 8-разрядных параллельных программируемых канала ввода-вывода. Режимы: ввод и вывод данных по двум 8-разрядным каналам, организация системы прерываний, счет импульсов, сдвиг вправо, преобразование параллельного кода в последовательный и обратно	48; (48 и 25)
К1809ВВ2 — системный последовательный канал	Четырехпроводная последовательная магистраль для организации межмашинного обмена. Максимальная длина магистрали витой парой до 5 м, коаксиальным кабелем до 100 м, подключается до 16 устройств	48; (48 и 25)
К1809ВГ1 — микроконтроллер магнитофона	Два режима работы: — обеспечивает управление бытовым магнитофоном в режиме внешнего запоминающего устройства — используется в системах обмена цифровой информацией по телефонным каналам со скоростью 50 ... 9600 бод	48; (48 и 25)
К1809ВГ3 — микроконтроллер телевизора и клавиатуры	Обеспечивает преобразование цифровых сигналов в видеосигналы телевизора и обратно. Изображение: до 16 цветов (тонов), знак 7×5 точек, 128 знаков в наборе, растр 256×256 точек. Обеспечивает управление клавиатурой с числом клавиш до 256	48; (48 и 25)
К531АП2П — двунаправленные магистральные усилители	Четыре двунаправленных магистральных усилителя, обеспечивающие согласование межплатной и внутриплатной магистралей и внутриплатное усиление линий на магистралях для обеспечения требуемого быстродействия микросистем	16; (16 и 8)

Примечание: БИС К1801ВП1-ХХХ выполняются на основе нескоммутированной вентильной матрицы. Конкретная схема получается нанесением на поверхность кристалла межэлементных связей, соответствующих заданной функции.

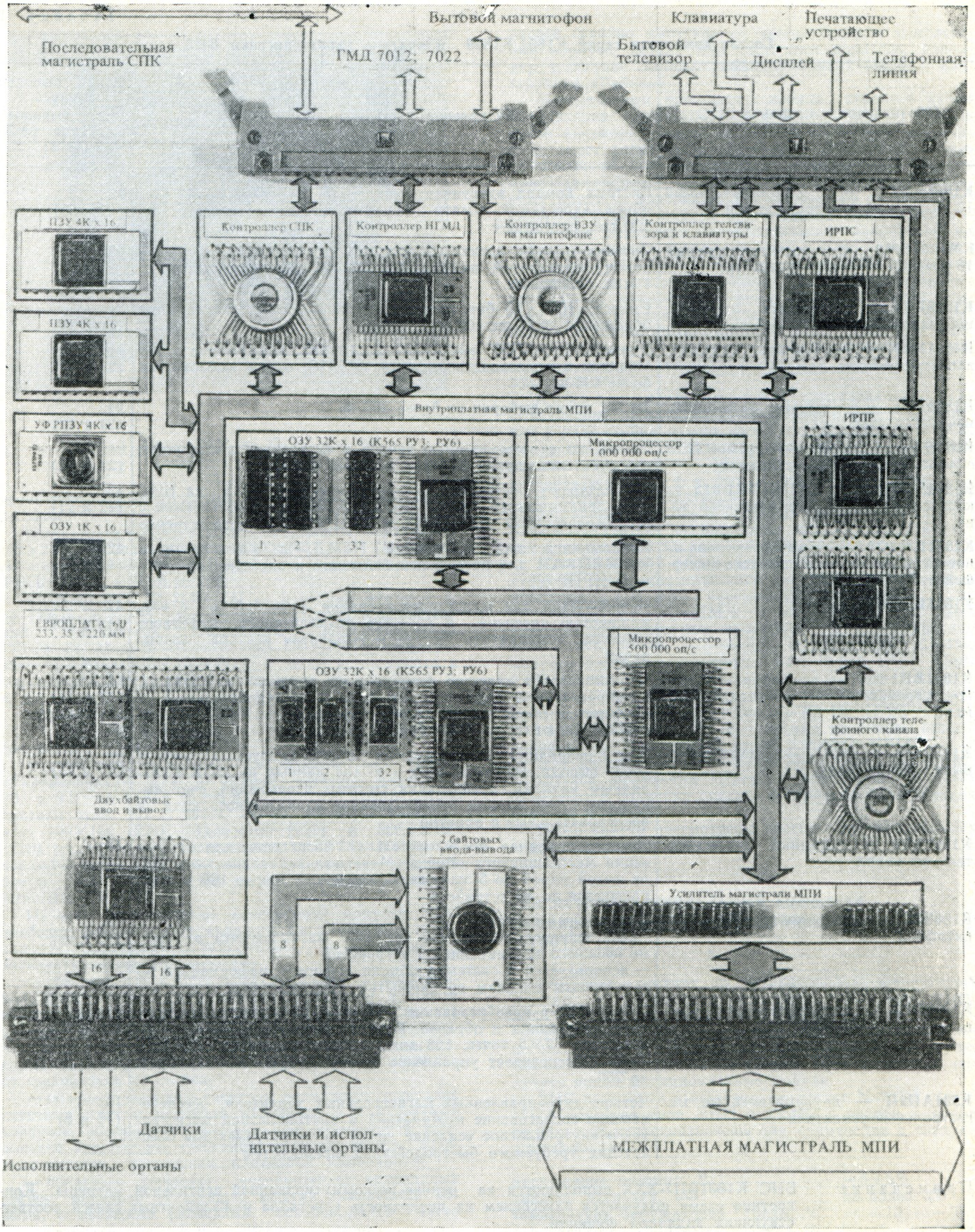


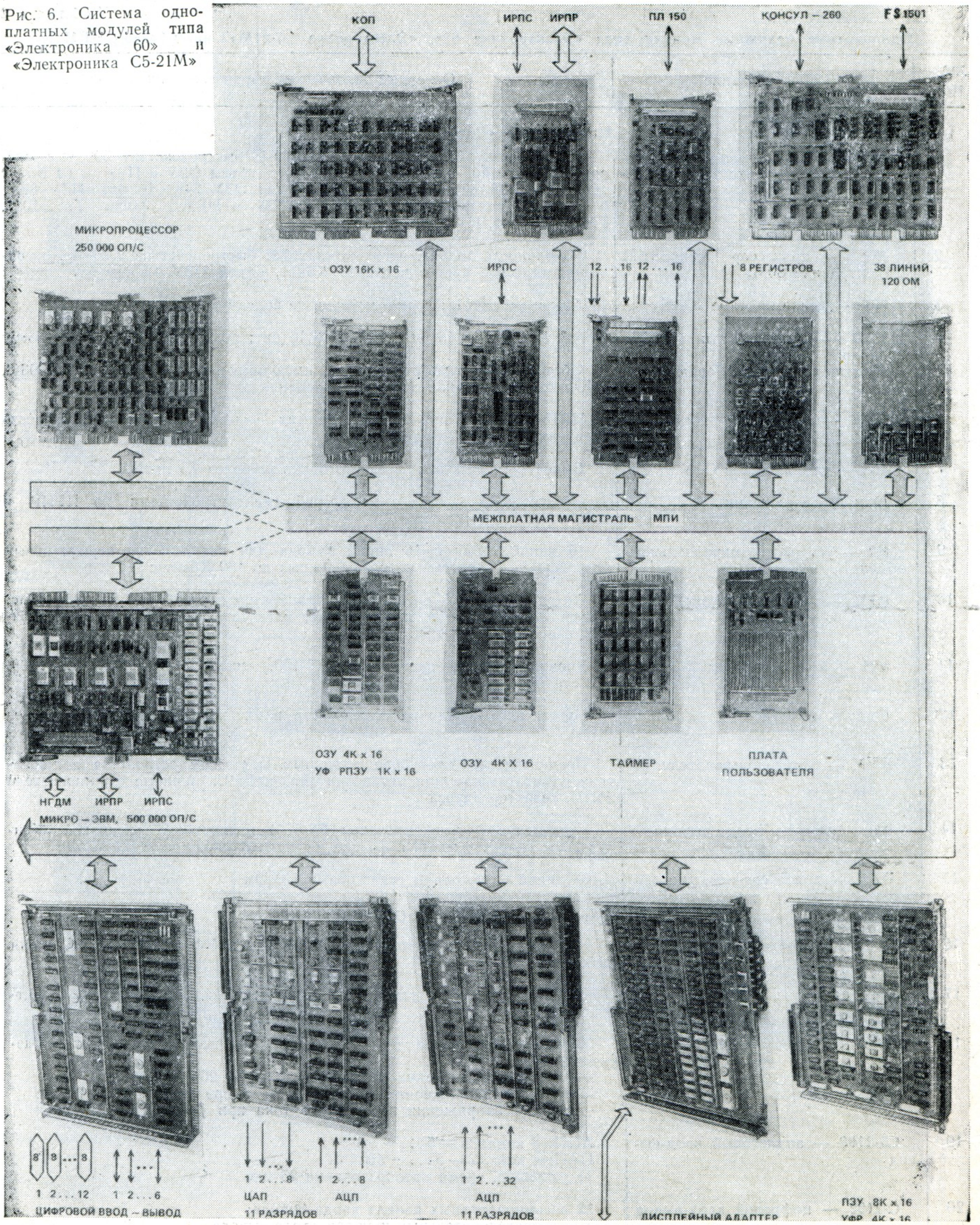
Рис. 5. Система однокристальных модулей серий К1801/К1809

Таблица 2

Одноплатные вдвижные модули типа «Электроника 60», «Электроника С5-21М» с интерфейсом МПИ

№№ п/п	Наименование	Основные технические характеристики
	Модули типа «Электроника 60»	
1	**МС 1201.01** — одноплатная микроЭВМ «Электроника НЦ-80-01Д»	Разрядность 16 бит; число команд 64; быстродействие 500 тыс. операций/с; адресное пространство 64 Кбайт; объем ОЗУ и ПЗУ — 64 Кбайт (ОЗУ 28К×16 бит+ПЗУ 4К×16 бит или ОЗУ 24К×16 бит+ПЗУ 8К××16 бит), контроллеры ИРПС, ИРПР и НГМД (типа ГМД-7012, ГМД-7022 и т. п.)
2	**М2** — процессор микроЭВМ «Электроника 60М»	Разрядность 16 бит; число команд 72; быстродействие 250 тыс. операций/с; адресное пространство 64 Кбайт; объем ОЗУ 4К×16 бит
3	**П3** — динамическое ОЗУ	Емкость 16К×16; время выборки не более 0,2 мкс
4	**ПЗП** — плата памяти и согласования	УФ РПЗУ 1К×16 бит; ОЗУ 4К×16 бит; терминатор и сетевой таймер
5	**Модуль связи с приборами**	Обеспечивает связь с приборами, имеющими интерфейс по ГОСТ 26.003-8, (МЭК 625-1) — канал общего пользования
6	**ПВК** — устройство последовательного обмена	ИРПС; режим дуплексный; формат посылки переменный; скорость передачи 75...9600 бод
7	**П2** — динамическое ОЗУ	Емкость 4К×16 бит; время выборки не более 0,5 мкс
8	**Модуль сопряжения**	Обеспечивает связь с периферийными устройствами по ИРПР и ИРПС
9	**И2** — устройство параллельного обмена	16 линий ввода и 16 линий вывода для подключения периферийных устройств со скоростью передачи данных до 180 Кбайт/с
10	**ПТЦ** — программируемый таймер	Интервалы от 2,8 до 2^{16} с; 6 поддиапазонов от 20 мкс до $200 \cdot 10^6$ мс; запуск и останов выполнения задач; измерение времени выполнения задач, счет событий и т. п.
11	**В21** — модуль вывода	Управление перфоратором ПЛ-150, скорость вывода 75; 100; 150 строк/с
12	**Модуль регистров**	Прием информации на 8 регистров и их программный опрос через МПИ
13	**ПИК** — плата пользователя	Предназначена для сбора объемным монтажом устройства пользователя. Имеет места для установки ИС и БИС, шинные формирователи МПИ, селектор адреса
14	**В1** — модуль ввода-вывода	Управление фотосчитывателем FS-1501 и пишущей машинкой «Консул 260»; скорость ввода и вывода не менее 8 знаков/с
15	**ПВЦ** — плата согласования	38 линий согласованы резисторами 120 Ом
	Модули типа «Электроника С5-21М»	
16	**С5-2113** — постоянное ЗУ	ПЗУ 8К×16 бит, время выборки 2,4 мкс и УФ РПЗУ 4К×16, время выборки 1 мкс
17	**С5-2106** — дисплейный адаптер	Управляет видеоконтрольным устройством типа «ВК 23В60», «Электроника К2-01» и т. п.
18	**С5-2101** — модуль АЦП	32 аналоговых входа (—5...+5В). Время преобразования в 11-разрядный код: — сигналов напряжения постоянного тока — 200 мкс, — сигналов напряжения переменного тока при $f=50$ Гц — 10 мс, — сигнала напряжения переменного тока при $f=400$ Гц — 1,25 мс,
19	**С5-2109** — аналоговый ввод-вывод	АЦП: 8 каналов — 5...+5В ЦАП: 4 канала — 5...+5В 11 разрядов, время преобразования 25 мкс
20	**С-2102** — цифровой ввод-вывод	12 восьмиразрядных канала ввода-вывода; шесть последовательных каналов ввода-вывода

Рис. 6. Система одноплатных модулей типа «Электроника 60» и «Электроника С5-21М»



атмосферное давление, кПа . . .	от 0,66 (5 мм рт. ст.) до 0,27 МПа (3 кгс/см²)

Состав комплекта приведен в табл. 1, а упрощенная схема подключения БИС к системной магистрали типа МПИ и внешним каналам — на рис. 5. Отдельные БИС описаны в работах [1, 4, 6, 7], остальные будут рассмотрены в последующих номерах журнала. Рис. 5 не следует рассматривать как вариант реальной одноплатной микроЭВМ. Он иллюстрирует лишь способ объединения однокристальных модулей при построении одноплатных микроЭВМ и систем. В реальном изделии нет необходимости в полной номенклатуре БИС комплекта, а кроме того, для него потребуется некоторое число микросхем с более низкой степенью интеграции и других ИЭТ, например, для формирования параметров выходных сигналов.

Система вдвижных модулей типа «Электроника 60» и «Электроника С5-21М»

Вдвижные модули типа «Электроника 60» и «Электроника С5-21М» получили наиболее широкое распространение в стране, выпускаются предприятиями различных министерств и ведомств. В их состав входят одноплатные микроЭВМ, микропроцессоры, модули запоминающих устройств, контроллеры для подключения ПУ, ВЗУ, УСО и т. д. Некоторые из выпускаемых в различных отраслях модули и способы их подключения к системной магистрали МПИ приведены на рис. 6, а краткие технические характеристики в табл. 2.

Таким образом, построение МСС на основе магистрально-модульного принципа обеспечивает возможность их широкого внедрения в различные отрасли народного хозяйства. Унификация конструкции, интерфейсов и программного обеспечения МСС, кроме явного технико-экономического эффекта, приносит еще и косвенную пользу, оказывая унифицирующее влияние на управляемые объекты, т. е. на машины, приборы и оборудование, выпускаемые различными отраслями народного хозяйства.

ЛИТЕРАТУРА

1. Одноплатные микроконтроллеры «Электроника С5-41» /М. П. Гальперин, А. В. Гинтер, В. В. Городецкий и др. — Микропроцессорные средства и системы, 1984, № 2, с. 20—23.

2. Мячев А. А., Никольский О. А. Стандартные интерфейсы микропроцессорных систем. — Микропроцессорные средства и системы, 1984, № 1, с. 27—33.

3. Никольский О. А. Перспективные модульные системы и локальные сети. — Микропроцессорные средства и системы, 1984, № 2, с. 91—92.

4. Гальперин М. П. Одноплатные микроЭВМ и микроконтроллеры. — Микропроцессорные средства и системы, 1984, № 2, с. 16—19.

5. Бабиченко С. И., Власкин А. Б., Гриценко А. Ф., Никольский О. А. Одноплатные процессорные устройства в Евростандарте на БИС серии К1801. — В кн.: Вопросы атомной науки и техники. Сер. Ядерное приборостроение. Вып 2(57). — М.: НИИП, ГКИАЭ, 1984, с. 41—46.

6. Лашевский Р. А., Тенк Э. Э., Хорин В.С. Однокристальное статическое ОЗУ со встроенным интерфейсом. — Микропроцессорные средства и системы, 1984, № 2, с. 85—87.

7. Однокристальные микропроцессоры комплекта БИС серии К1801/К1809 /В. Л. Дшхунян, Ю. И. Борщенко, В. Р. Науменков и др. — Микропроцессорные средства и системы, 1984, № 4, с. 12—18.

От редакции. Общее число разработанных и выпускаемых серийно модулей типа «Электроника 60», т. е. совместимых по интерфейсу, конструкции и программному обеспечению с микроЭВМ «Электроника 60, -60М, -60-1, НЦ-80-01Д, НЦ-80-01ДМ» и т. д. велико, но полной информации о них нет. По просьбе читателей редакция обращается к разработчикам с предложением сообщить информацию о подобных модулях в форме прилагаемой анкеты.

По материалам лучших разработок будет подготовлен развернутый обзор.

Анкета

1. Условное обозначение
2. ТУ (при наличии)
3. Наименование
4. Предприятие-разработчик (подробный адрес)
5. Предприятие-изготовитель (при наличии)
6. Технические характеристики:
   6.1. Общие
      6.1.1. Напряжение питания, В
      6.1.2. Потребляемая мощность, В·А
      6.1.3. Габаритные размеры, мм
      6.1.4. Масса, кг
      6.1.5 Диапазон рабочих температур, °С
      6.1.6. Цена, руб.
   6.2. Показатели функционирования (приводится 4...6 основных для данного изделия показателей, желательно по ОСТ 11 305.912.0-14-83)

7. Структурная схема модуля (6...10 блоков) с выявлением способа подключения к МПИ (располагать слева по чертежу) и связи к внешним относительно модуля устройствам, если такие имеются (располагать справа по чертежу).

8. Краткое описание функционирования с отражением взаимодействия модуля с внешними устройствами.

9. Фотонегатив модуля, черно-белый, размером 24×36 мм, вид со стороны микросхем с размещением объектива над центром платы, и две фотографии 13×18 см с указанием на обратной стороне (карандашом) наименования изделия.

Общий объем материала на одно изделие — 5 стр. машинописного текста. Авторов просим указать адрес и номер телефона.

Статья поступила 17 октября 1984 г.

---

29 международный симпозиум: «Мини- и микроЭВМ и их приложения»

Июнь 25—28, 1985 г., Испания (Sant Feliu de Guixols, Gerona)

Организатор: Международное общество мини- и микроЭВМ (IMM).

Тематика: Перекрывает все аспекты мини- и микроЭВМ с особым вниманием к практическим приложениям и включает:

технологию, аппаратуру, СБИС, САПР, графику, анализ технических наук (с точки зрения МП-приложений), системы автоматизированного тестирования и диагностики, системы разработки (MDS);

АЛУ, интерфейсные средства, средства межмашинной связи, распределенные системы, многопроцессорные системы, системы, устойчивые к отказам, системы реального времени, технология программного обеспечения, базы данных;

производственная технология, измерения, анализ сигналов и изображений, роботы и системы автоматического управления, источники питания, здравоохранение и биотехнология, системы автоматизации учреждений и банков, образование.

УДК 681.323—181.48

В. Л. Дшхунян, Ю. И. Борщенко, В. Р. Науменков,
А. А. Рыжов, Ю. В. Романец, И. А. Бурмистров, Е. М. Соловьев

ОДНОКРИСТАЛЬНЫЕ МИКРОПРОЦЕССОРЫ КОМПЛЕКТА БИС СЕРИИ К1801

Однокристальные микропроцессоры (ОМП) К1801ВМ1 и КМ1801ВМ2 являются основой магистрально-модульного микропроцессорного комплекта БИС серий К1801/К1809 [1, 2], обеспечивающего возможность построения широкого класса микроЭВМ, различных микроконтроллеров и управляющих микропроцессорных систем.

Однокристальный микропроцессор К1801ВМ1

Микросхема К1801ВМ1 является однокристальным 16-разрядным микропроцессором, предназначенным для обработки цифровой информации в системах управления технологическими процессами, контрольно-измерительной аппаратуре и системах связи, а также решения в составе ЭВМ инженерно-технических и экономических задач.

Микропроцессор содержит следующие основные функциональные блоки, объединенные информационно-управляющими связями (рис. 1):

16-разрядный операционный блок, выполняющий операции формирования адресов команд и операндов, логические и арифметические операции, хранение операндов и результатов;

блок микропрограммного управления, вырабатывающий последовательность микрокоманд на основе кода принятой команды. Выполнен на базе программируемой логической матрицы (ПЛМ), содержащей 250 логических произведений. В ПЛМ закодирован полный набор микрокоманд для всех типов команд. Программирование ПЛМ производится на этапе изготовления микросхемы с помощью одного фотошаблона;

блок прерываний, организующий приоритетную систему прерываний. Выполняет прием и предварительную обработку внешних и внутренних запросов на прерывание вычислительного процесса;

интерфейсный блок, отвечающий за обмен информацией между ОМП и устройствами, расположенными на системной магистрали. Осуществляет арбитраж при операциях прямого доступа к памяти, формирует последовательность управляющих сигналов;

блок системной магистрали, связывающий внутреннюю магистраль ОМП с внешней. Управляет усилителями приема и выдачи информации на совмещенные выводы адресов и данных;

схема тактирования, обеспечивающая синхронизацию внутренних блоков ОМП.

Ниже приведены основные технические характеристики ОМП.

Система команд (табл. 1), реализованная в ПЛМ блока микропрограммного управления ОМП соответствует микроЭВМ «Электроника 60».

Помимо перечисленных команд ОМП имеет в своем составе специальные команды, предназначенные для работы с системным ПЗУ К1801РЕ1.

Условно-графическое обозначение микросхемы ОМП и наименование выводов приведены на рис. 2. Сигналы AD0...AD15 определяют адреса и данные, которые передаются по совмещенной системной магистрали. Передача адресов и данных по одним и тем же выводам обеспечивается разделением во времени.

Группа сигналов SYNC, DIN, DOUT, WTBT, RPLY управляет передачей информации по системной магистрали. Сигнал SYNC, вырабатываемый процессором, означает, что адрес находится на выводах системной магистрали. Этот сигнал сохраняет активный уровень до окончания текущего обмена информацией.

Сигнал RPLY означает, что данные приняты или установлены на информационных выводах. Этот сигнал вырабатывается пассивным устройством в ответ на сигналы DIN и DOUT. Сигнал DIN предназначен для организации двух процедур обмена информацией по магистрали:

ввода данных — ОМП вырабатывает DIN во время действия сигнала SYNC, когда он готов принять данные от пассивного устройства;

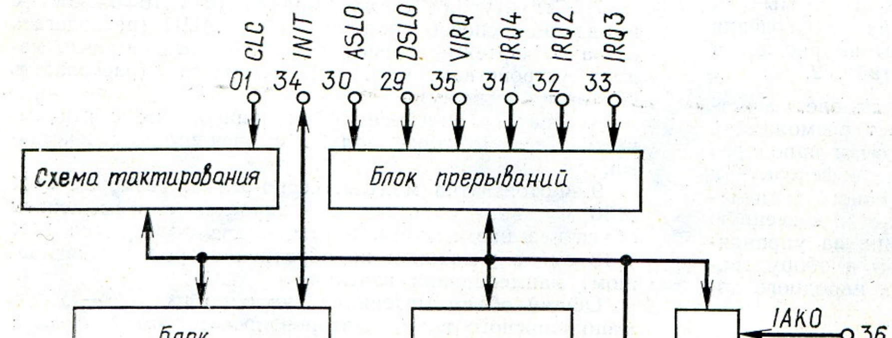


Рис. 1. Структурная организация процессора К1801ВМ1

Технические характеристики микропроцессора К1801ВМ1

Представление чисел	дополнительный код с фиксированной запятой
Система команд	безадресная, одноадресная, двухадресная
Виды адресации	регистровая, косвенно-регистровая, автоинкрементная, косвенно-автоинкрементная автодекрементная, косвенно-автодекрементная, индексная, косвенно-индексная
Число регистров общего назначения (РОН)	8
Число уровней прерываний	4
Системная магистраль	типа МПИ
Адресное пространство, Кбайт	64
Тактовая частота, МГц	до 5
Максимальное быстродействие при выполнении регистровых операций, операций/с	до 500 тыс.
Потребляемая мощность, Вт	до 1
Напряжение питания, В	+5±5%
Уровни логических сигналов, В:	
«Лог. 0» (активный уровень)	менее 0,5
«Лог. 1»	более 2,4
Нагрузочная способность по току, мА	3,2
Емкость нагрузки, пФ	до 100
Технология изготовления	N-МОП
Конструкция	42-выводной планарный металлокерам. корпус

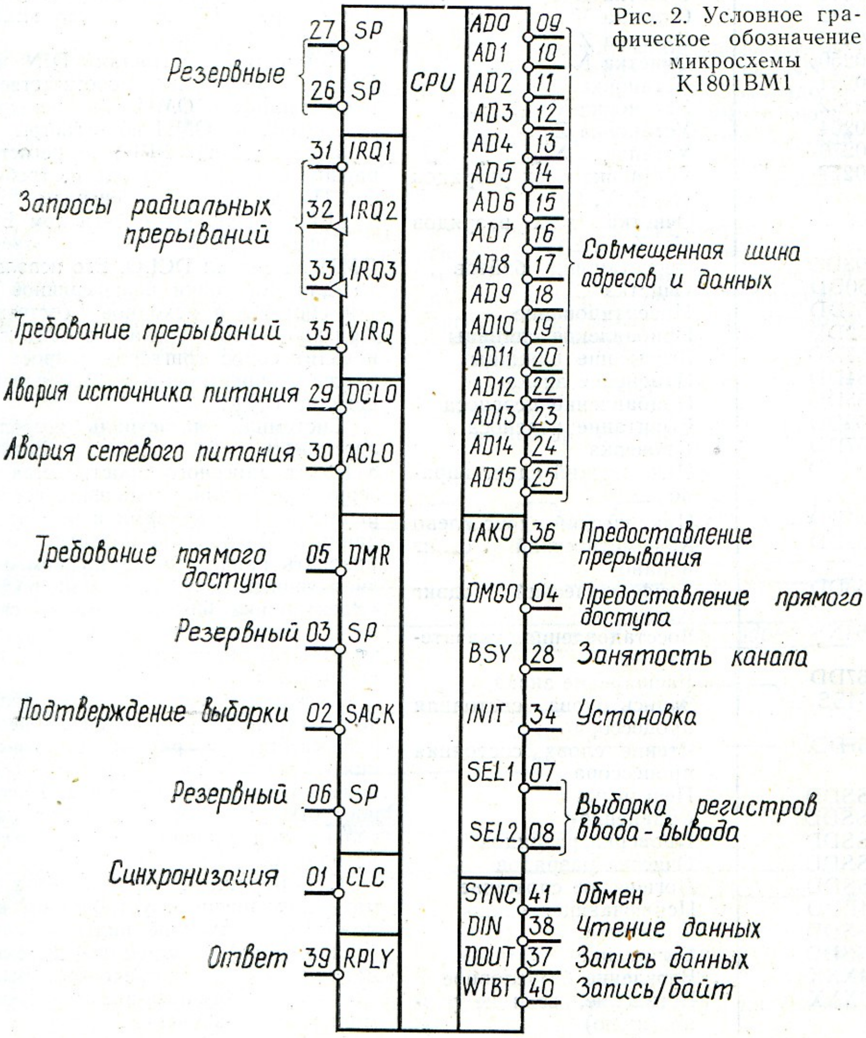


Рис. 2. Условное графическое обозначение микросхемы К1801ВМ1

ввода адреса вектора прерывания — сигнал DIN вырабатывается совместно с сигналом IAKO при пассивном уровне сигнала SYNC.

Сигнал DOUT означает, что данные, выдаваемые ОМП, установлены на выводах системной магистрали.

Сигнал WTBT предназначен для организации двух процедур обмена информацией:

вырабатывается в адресной части цикла для указания о том, что далее следует вывод данных (слова или байта);

формируется при выводе данных из ОМП для указания о выводе байта.

Сигнал VIRQ вырабатывается внешним устройством для информирования ОМП о том, что оно готово передавать адрес вектора прерывания. В ответ на этот сигнал, если прерывание разрешено, ОМП вырабатывает сигналы DIN и IAKO.

Сигнал IRQ1 определяет положение внешнего переключателя «Программа — пульт». Низкий уровень этого сигнала означает, что переключатель находится в положении «Пульт». Этот сигнал переводит ОМП в состояние, аналогичное состоянию после выполнения команды HALT.

Сигналы IRQ2 и IRQ3 вызывают прерывание программы, выполняемой процессором, по фиксированным адресам 000100_8 и 000270_8 соответственно. Прерывание происходит при переходе сигналов из высокого уровня в низкий.

Сигнал предоставления прерываний IAKO процессор вырабатывает в ответ на внешний сигнал VIRQ. Этот сигнал является выходным для ОМП и входным для первого устройства, подключенного к системной магистрали (электрически ближе расположенного к ОМП, и, следовательно, имеющего более высокий приоритет). Если это устройство не требовало прерываний (не устанавливало сигнал VIRQ), то оно транслирует сигнал IAKO к следующему устройству. Устройство, требующее прерывание ОМП, запретит распространение этого сигнала. Сигнал IAKO, последовательно проходя через все устройства, обеспечивает их поочередный опрос и различный приоритет обслуживания.

Сигнал DMR вырабатывает внешнее активное устройство, требующее передачи ему системной магистрали. Сигнал DMGO процессор устанавливает в ответ на внешний сигнал DMR. Этот сигнал последовательно проходит через внешние устройства и предоставляет системную магистраль устройству с наивысшим приоритетом, запросившему прямой доступ к памяти. Это устройство прекращает трансляцию сигнала DMGO и устанавливает сигнал SACK, который вырабатывается устройством прямого доступа к памяти (ПДП) в ответ на сигнал DMGO. Сигнал SACK означает,

Таблица 1

Система команд микропроцессора К1801ВМ1

№№ пп.	Обозначение	Код	Наименование
1	2	3	4
1	HALT	000000	Останов
2	WAIT	000001	Ожидание
3	RTI	000002	Возврат из прерывания
4	BPT	000003	Командное прерывание для отладки
5	IOT	000004	Командное прерывание для ввода-вывода
6	RESET	000005	Сброс внешних устройств
7	RTT	000006	Возврат из прерывания
8	JMP	0001DD	Безусловный переход
9	RTS	00020R	Возврат из подпрограммы
10	JSR	004RDD	Обращение к подпрограмме
11	EMT	104000... ...104377	Командное прерывание для системных программ
12	TRAP	104400... ...104777	Командное прерывание
13	NOP	000240	Нет операции
14	CLC	000241	Очистка C
15	CLV	000242	Очистка V
16	CLZ	000244	Очистка Z
17	CLN	000250	Очистка N
18	SEC	000261	Установка C
19	SEV	000262	Установка V
20	SEZ	000264	Установка Z
21	SEN	000270	Установка N
22	SCC	000277	Установка всех разрядов (N, Z, V, C)
23	CCC	000257	Очистка всех разрядов (N, Z, V, C)
24	SWAB	0003DD	Перестановка байтов
25	CLR (B)	*050DD	Очистка
26	COM (B)	*051DD	Инвертирование
27	INC (B)	*052DD	Прибавление единицы
28	DEC (B)	*053DD	Вычитание единицы
29	NEG (B)	*054DD	Изменение знака
30	ADC (B)	*055DD	Прибавление переноса
31	SBC (B)	*056DD	Вычитание переноса
32	TST (B)	*057DD	Проверка
33	ROR (B)	*060DD	Циклический сдвиг вправо
34	ROL (B)	*061DD	Циклический сдвиг влево
35	ASR (B)	*062DD	Арифметический сдвиг вправо
36	ASL (B)	*063DD	Арифметический сдвиг влево
37	MAPK	0064NN	Восстановление указателя стека
38	SXT	0067DD	Расширение знака
39	MTPS	1064SS	Запись слова состояния процессора
40	MFPS	1067DD	Чтение слова состояния процессора
41	MOV (B)	*1SSDD	Пересылка
42	CMP (B)	*2SSDD	Сравнение
43	BIT (B)	*3SSDD	Проверка разрядов
44	BIC (B)	*4SSDD	Очистка разрядов
45	BIS (B)	*5SSDD	Логическое сложение
46	XOR	074RDD	Исключающее «или»
47	ADD	06SSDD	Сложение
48	SUB	16SSDD	Вычитание
49	BR	0004XXX	Ветвление безусловное
50	BNE	0010XXX	Ветвление, если не равно (нулю)

что устройство ПДП может производить обмен данными, используя стандартные циклы обращения к системной магистрали.

Сигнал занятости магистрали BSY предназначен для управления устройствами умощнения магистрали. Низкий уровень этого сигнала означает, что ОМП начинает обмен по магистрали. Переход сигнала из низкого уровня в высокий означает окончание обмена.

Сигнал аварии источника питания DCLO вызывает установку ОМП в исходное состояние и появление сигнала INIT. Сигнал аварии сетевого питания ACLO означает переход ОМП на обработку прерывания по сбою питания. Высокий уровень этого сигнала свидетельствует о том, что сетевое напряжение питания нормально.

При появлении одного из сигналов обращения к внешним регистрам расширения ввода-вывода SEL1, SEL2 адрес на выводах системной магистрали соответствует адресу одного из внешних регистров расширения ввода-вывода. Сигнал SEL1 устанавливается при обращении по фиксированному адресу 177716_8, сигнал SEL2 — по адресу 177714_8.

Совместно с сигналами DIN или DOUT происходит соответственно ввод данных в ОМП из регистров или вывод из ОМП на регистры. Установка сигнала RPLY от регистров расширения ввода-вывода не требуется. По длительности сигналы SEL1 и SEL2 совпадают с сигналом BSY.

Сигнал INIT является ответом ОМП на сигнал DCLO. Его используют для установки периферийной части системы в исходное состояние. При вводе этого сигнала в ОМП происходит сброс триггеров запроса радиальных прерываний и блокирования сигнала DMR.

Системная магистраль позволяет адресовать 64 Кбайта. Верхние 8 Кбайт адресного пространства резервируются для управления периферийными устройствами и регистрами данных. Магистраль позволяет организовать канал обмена информацией, аналогичный каналу микроЭВМ «Электроника 60», в котором связь между двумя устройствами осуществляется по принципу «активный-пассивный».

Активное устройство управляет прохождением информации по системной магистрали, разрешает прерывания, обеспечивает предоставление прямого доступа к памяти. Пассивное устройство передает информацию только под управлением активного устройства.

При обращении процессора к памяти или внешним устройствам возможны следующие виды обменов (циклов) информацией по системной магистрали: Ввод (Чтение), Вывод (Запись), Ввод-пауза-вывод (Чтение-модификация-запись).

Продолжение табл. 1

1	2	3	4
51	BEQ	0014XXX	Ветвление, если равно (нулю)
52	BGE	0020XXX	Ветвление, если больше и равно (нулю)
53	BLT	0024XXX	Ветвление, если меньше (нуля)
54	BGT	0030XXX	Ветвление, если больше (нуля)
55	BLE	0034XXX	Ветвление, если меньше или равно (нулю)
56	SOB	077RNN	Вычитание единицы и ветвление
57	BPL	1000XXX	Ветвление, если плюс
58	BMI	1004XXX	Ветвление, если минус
59	BHI	1010XXX	Ветвление, если больше
60	BLOS	1014XXX	Ветвление, если меньше или равно
61	BVC	1020XXX	Ветвление, если нет арифметического переполнения
62	BVS	1024XXX	Ветвление, если арифметическое переполнение
63	BHIS, BCC	1030XXX	Ветвление, если больше или равно
64	BL0, BCS	1034XXX	Ветвление, если меньше

Примечание: B — байтовые команды; SS — поле адресации операнда источника; DD — поле адресации операнда приемника; NN — смещение (6 разрядов); XXX — смещение (8 разрядов); R — регистр общего назначения (РОН); * — имеет значение: 0 — для команд с операцией над словами; 1 — для команд с операцией над байтами.

Цикл Ввод-пауза-вывод включает ввод данных, выполнение арифметическо-логических операций и вывод результата без повторения передачи адреса, т. е. результат записывается по адресу последнего выбранного операнда.

Ниже иллюстрируются некоторые виды обмена информацией по системной магистрали. При выполнении цикла Ввод данные передаются от пассивного устройства к активному. Временная диаграмма цикла приведена на рис. 3. Черта над обозначением сигнала указывает, что активным уровнем является низкий уровень сигнала.

Порядок выполнения операций следующий:

процессор на выводах системной магистрали устанавливает адрес и выдает сигнал BSY;

синхронно с выдачей и снятием адреса ОМП устанавливает и снимает сигнал синхронизации SYNC, по которому происходит запоминание адреса пассивным устройством;

после установки сигнала SYNC ОМП снимает адрес, устанавливает сигнал DIN и ожидает поступления сигнала RPLY от пассивного устройства. Если в течение 64 тактов частоты синхронизации сигнал RPLY не появился, ОМП переходит к обслуживанию внутреннего прерывания по ошибке обращения к системной магистрали;

после приема сигнала RPLY ОМП принимает данные от пассивного устройства и снимает сигнал DIN;

после снятия сигнала DIN пассивное устройство снимает сигнал RPLY, завершая операцию передачи данных;

после снятия сигнала RPLY пассивным устройством ОМП снимает сигналы SYNC и BSY;

если системная магистраль не предоставляется устройству ПДП, то процессор сразу после снятия сигнала SYNC может начать новый цикл обращения к магистрали. В этом случае сигнал BSY не снимается. Во время цикла Ввод сигнал WIBT не вырабатывается.

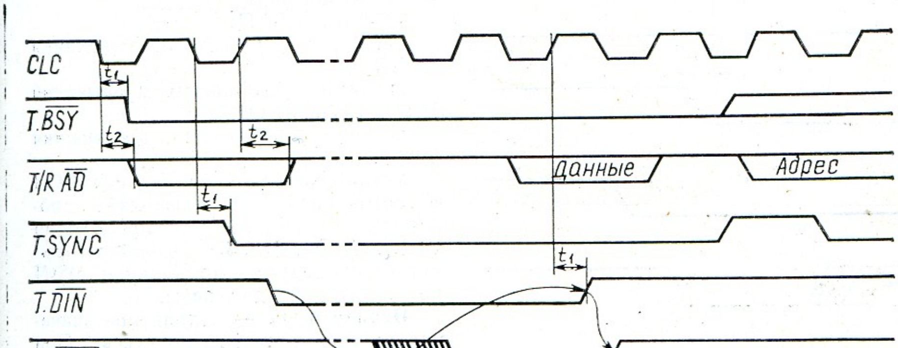


Рис. 3. Временна́я диаграмма цикла Ввод (Чтение) информации по параметрам: Т — передатчик; Р — приемник; t_1=40...70 нс; t_2=50...80 нс; * — область прихода сигнала $\overline{RPLY}$, для которой соблюдается приведенная диаграмма

Кроме ОМП, **активным устройством на системной магистрали может быть устройство ПДП**, которое в режиме ПДП выполняет адресацию, синхронизацию, вырабатывает управляющие сигналы для организации стандартных циклов обращения к системной магистрали без вмешательства ОМП. Временная диаграмма ПДП представлена на рис. 4. Устройство ПДП устанавливает сигнал DMR, в ответ на который ОМП формирует сигнал DMGO, затем, получив сигнал DMGO, вырабатывает сигнал SACK и снимает сигнал DMR. ОМП снимает сигнал DMGO и ожидает завершения операции ПДП. Устройство ПДП начинает выполнять циклы передачи данных, аналогичные циклам Ввод, Вывод или Ввод-пауза-вывод. После окончания обмена данными устройство ПДП снимает сигнал SACK, возвращая управление магистралью процессору.

Регистры общего назначения (РОН) используются в качестве индексных и накопительных, регистров автоинкрементной и автодекрементной адресаций. Среди восьми РОН два регистра R6 и R7 имеют специальное назначение: R6 используется в качестве указателя стека и содержит адрес последней ячейки стека; R7 является счетчиком команд (СК) и содержит адрес очередной выполняемой команды.

Кроме РОН, программно доступным регистром является регистр состояния процессора (РСП), содержащий информацию о текущем приори-

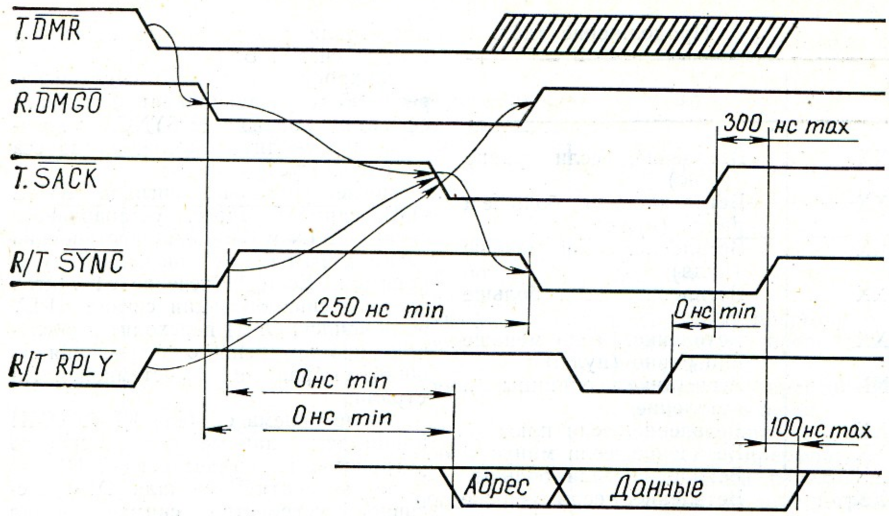


Рис. 4. Временная диаграмма цикла ПДП

тете ОМП, значения кодов условий ветвлений программы, состояние «Т-бита», используемого при отладке программы и вызывающего ее прерывание.

Прерывание — это временное прекращение выполнения текущей программы и переход к выполнению программы обслуживания устройства, вызвавшего прерывание. Каждое устройство, способное вызвать прерывание, должно иметь регистр состояния, содержащий разряд разрешения прерывания. Если устройству разрешается прерывание программы, то этот разряд должен быть программно установлен в единичное состояние. ОМП будет предоставлять прерывание и, следовательно, выполнять программу обслуживания устройства только в том случае, если седьмой разряд его РСП будет установлен в нулевое состояние. Каждое устройство, требующее прерывания, имеет программу обслуживания, вход в которую осуществляется автоматически с помощью вектора прерывания. Временная диаграмма приема адреса вектора прерывания представлена на рис. 5.

Последовательность операций при этом следующая:

устройство, которому необходимо обслуживание, выставляет сигнал требования прерывания VIRQ;

если прерывание разрешено, то процессор помещает в стек содержимое СК и РСП и последовательно устанавливает сигналы DIN и IAKO;

устройство принимает сигнал IAKO и запрещает распространение этого сигнала к другим устройствам, помещает адрес вектора прерывания на выводы системной магистрали, вырабатывает сигнал RPLY и снимает сигнал VIRQ;

процессор принимает адрес вектора прерывания и последовательно снимает сигналы DIN и IAKO;

устройство завершает передачу вектора и снимает сигнал RPLY.

ОМП загружает новое содержимое СК и РСП из двух последовательных ячеек, первая из которых определяется адресом вектора прерывания, после чего переходит к выполнению программы обслуживания данного устройства. После завершения программы обслуживания ОМП возобновляет выполнение прерванной программы с помощью команды возврата из прерывания, по которой из стека выбираются два слова и записываются в СК и РСП.

Если имеется несколько устройств, способных запрашивать прерывание, то они соединяются по цепи предоставления прерывания (рис. 6). Прерывание текущего процесса и последующая его обработка производится после выполнения очередной команды. Прерывание по ошибке обращения к системной магистрали (обращение по адресу несуществующей ячейки памяти или регистра внешнего устройства) может прервать выполнение программы на любой фазе исполнения команды.

Источниками прерываний, расположенными в соответствии с приоритетностью обработки (в случае одновременного появления нескольких запросов) являются:

ошибка обращения к магистрали;

резервный или запрещенный код в регистре команд;

«Т-бит» в РСП;

сигнал аварии сетевого питания ACLO;

сигналы радиальных прерываний IRQ1, IRQ2, IRQ3;

сигнал векторного прерывания VIRQ.

Установка седьмого разряда РСП в состояние «Лог.1» позволяет игнорировать сигналы прерывания VIRQ, IRQ2, IRQ3. Микропрограммная процедура прерывания ОМП разделяется на две фазы.

Первая фаза — сохранение значения вектора текущего процесса. При этом происходит запись в память по указателю стека (R6) с предварительным декрементом. Сначала записывается значение информации, содержащейся в РСП, затем — в СК.

Вторая фаза — загрузка значения вектора прерывания. ОМП формирует значение адреса вектора прерывания, по которому происходит загрузка СК

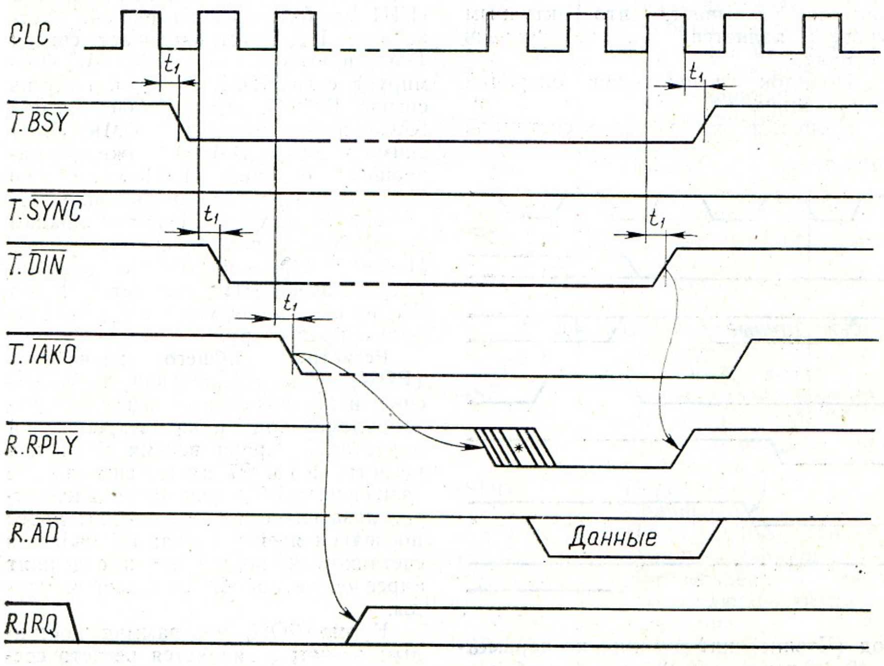


Рис. 5. Временная диаграмма процедуры приема адреса вектора прерывания

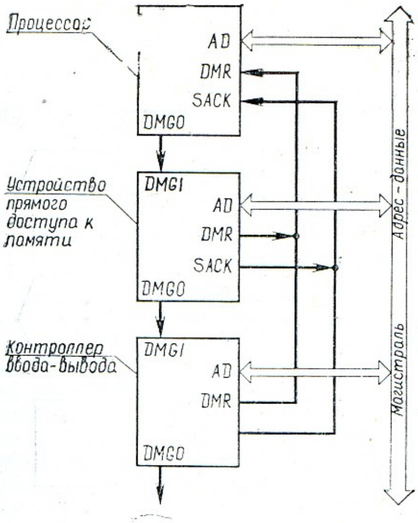


Рис. 6. Соединение нескольких устройств по цепи предоставления прерываний

и регистра слова состояния процессора (ССП) значениями двух ячеек памяти. ОМП анализирует состояние запросов на прерывание: если незамаскированных запросов нет, то происходит чтение первой команды нового процесса и ее выполнение. Иначе снова происходит процедура прерывания (рис. 7).

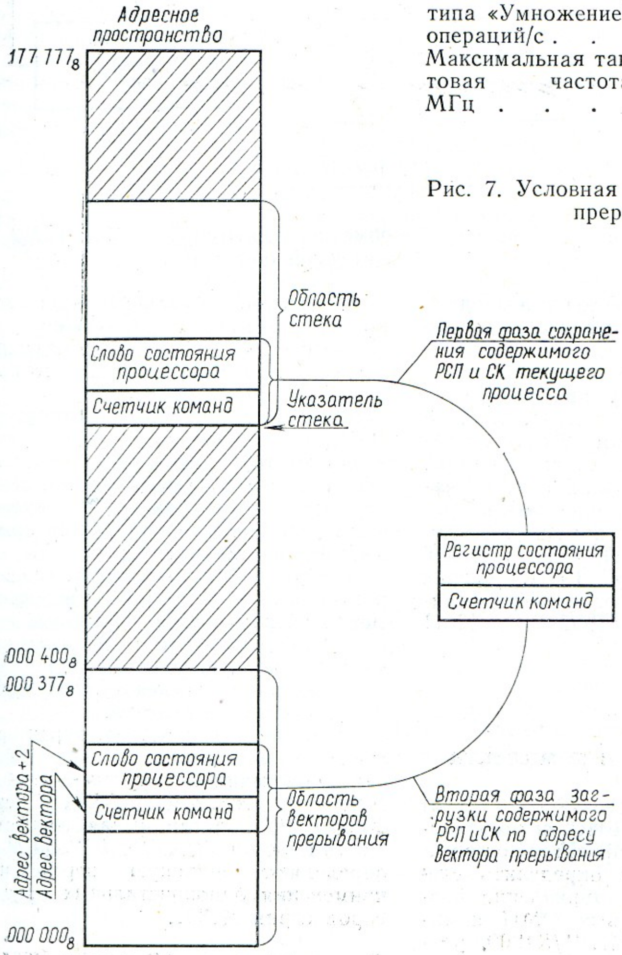


Рис. 7. Условная схема процедуры прерывания

Однокристальный микропроцессор КМ1801ВМ2

Однокристальный микропроцессор КМ1801ВМ2 является дальнейшим развитием ОМП К1801ВМ1, обладает более высоким быстродействием, расширенной системой команд и рядом новых функциональных возможностей при применении в микроЭВМ. Имеет запасы по всем параметрам, что позволяет повысить экономические характеристики и, в конечном счете, сделать микросхему доступной для самого широкого круга потребителей. Совместим по программному обеспечению и интерфейсу с ОМП К1801ВМ1. Основные параметры ОМП КМ1801ВМ2 приведены ниже.

Быстродействие выполнения регистровых команд типа «Сложение», операций/с . . . **до 1000 тыс.**
Быстродействие выполнения команд типа «Умножение», операций/с . . . . до 100 тыс.
Максимальная тактовая частота, МГц 10
Мощность потребления, Вт . . . до 1,7
Конструкция . . . металлокерамический 40-выводной корпус типа ДИП

По сравнению с микросхемой К1801ВМ1 процессор КМ1801ВМ2 имеет расширенную систему команд (табл. 2).

Команды расширенной арифметики (MUL, DIV, ASH, ASHC) реализованы на микропрограммном уровне в блоке микропрограммного управления ОМП. Выполнение команд с плавающей запятой (FADD, FSUB, FMUL, FDIV) обеспечивается на программном уровне с помощью системного ПЗУ К1801РЕ1.

Для повышения быстродействия выполнения последовательности команд в ОМП реализован конвейерный принцип приема и выполнения команд, заключающийся в том, что к концу выполнения i-й команды ($i+1$)-я команда уже принята и процессор начинает прием ($i+2$)-й команды.

Условное графическое обозначение приведено на рис. 8. По сравнению с К1801ВМ1 микросхема КМ1801ВМ2 имеет ряд функциональных особенностей.

Сигнал выходной тактовой синхронизации CLC0 имеет частоту, равную половине частоты входной синхронизации CLC1. Он предназначен для организации систем, обладающих повышенной надежностью, в которых несколько процессоров должны работать синхронно.

Микросхема КМ1801ВМ2 наряду с синхронной адресной частью обмена по системной магистрали имеет возможность асинхронной работы при передаче адреса. Это обеспечивается с помощью входного сигнала Адрес принят (AR). После выдачи адреса на системную магистраль ОМП ожидает появление сигнала AR. Низкий

Таблица 2

Дополнительные команды, реализуемые процессором КМ1801ВМ2

№№ пп.	Обозначение	Код	Наименование
1	MUL	070RSS	Умножение
2	DIV	071RSS	Деление
3	ASH	072RSS	Сдвиг на N разрядов одного слова
4	ASHC	073RSS	Сдвиг на N разрядов двойного слова
5	FADD	07500R	Сложение с плавающей запятой
6	FSUB	07501R	Вычитание с плавающей запятой
7	FMUL	07502R	Умножение с плавающей запятой
8	FDIV	07503R	Деление с плавающей запятой

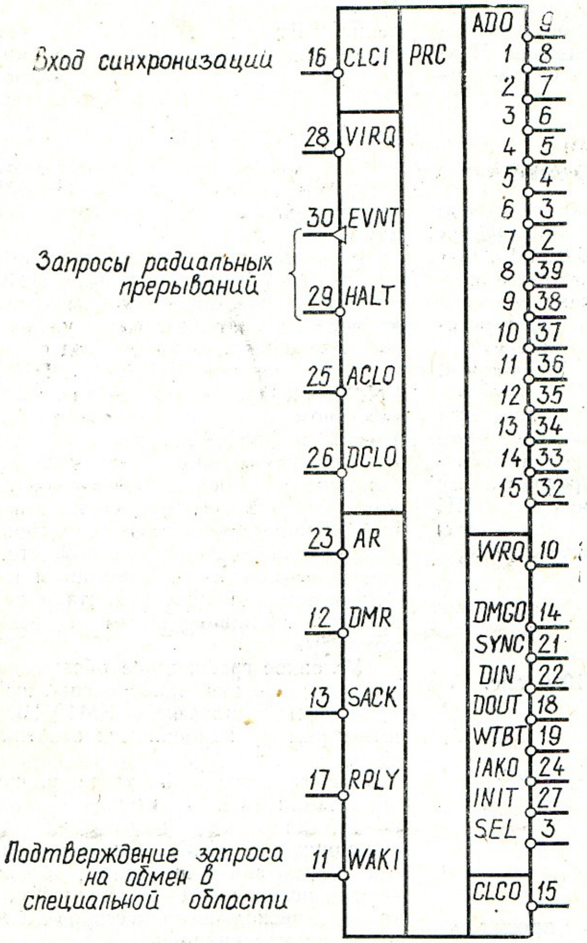


Рис. 8. Условное графическое обозначение микросхемы процессора КМ1801ВМ2

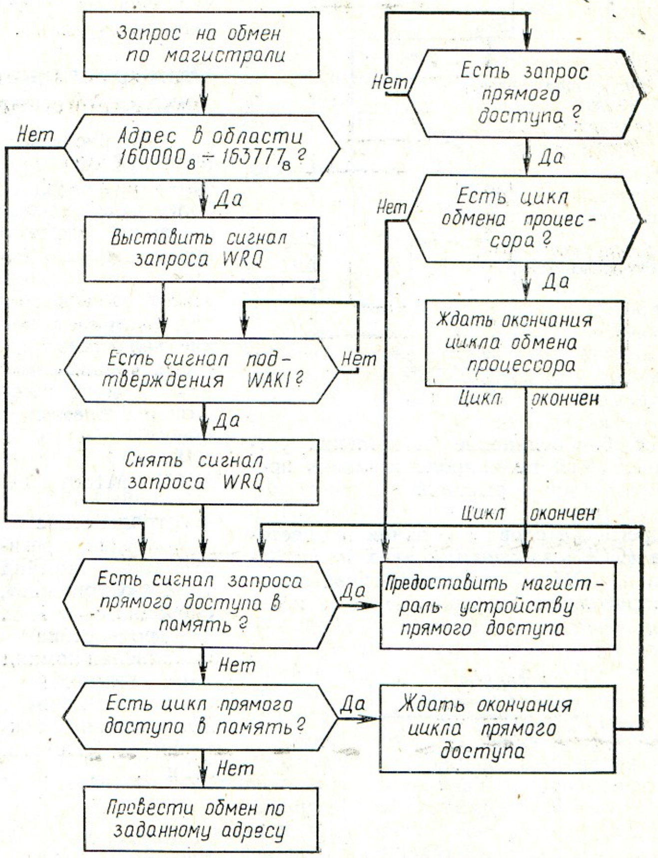


Рис. 9. Процедура обмена процессора КМ1801ВМ2 в специальной области

уровень входного сигнала AR свидетельствует о том, что пассивное устройство приняло адрес и ОМП может продолжать цикл обмена по системной магистрали.

ОМП обладает встроенной логикой обращения в область адресного пространства от 160000_8 до 163777_8 включительно по специальной процедуре, представленной на рис. 9. Описанный механизм обращения в специальную область использует сигналы WRQ и WAKI и может применяться для организации обмена в многопроцессорных вычислительных системах.

Сигнал SEL, устанавливаемый процессором в фазе чтения одновременно с сигналом DIN, определяет процедуру чтения внешнего регистра ввода-вывода. Обычно таким регистром является регистр начальных условий пуска процессора в составе ЭВМ. Если сигнал SEL ОМП устанавливает в фазе выдачи адреса, то это означает, что обмен идет с системной памятью, в качестве которой используется микросхема К1801РЕ1. Системная память помимо специальных программ (начального пуска, резидентного тестового обеспечения, пультового терминала) позволяет расширять систему команд с помощью программной реализации дополнительных команд.

Системная память при работе с ОМП является «скрытой», т. е. ее адреса в программном режиме не занимают адресного пространства ОЗУ. Тем самым она дополняет оперативную память. Системная память программно доступна с помощью специальных команд ОМП. Сигнал HALT аналогичен сигналу IRQ1, сигнал EVNT (прерывание от таймера) аналогичен сигналу IRQ2 в ОМП К1801ВМ1.

Развитие однокристальных микропроцессоров комплекта

Применение перспективной технологии изготовления БИС, автоматизированных систем проектирования и контроля позволяет определить следующие основные направления развития однокристальных ОМП комплекта БИС серий К1801/К1809, реализуемых в настоящее время:

увеличение быстродействия выполнения команд в 2,3 раза по сравнению с ОМП КМ1801ВМ2, расширение системы команд и адресного пространства до 4 Мбайт;

совершенствование структуры ОМП, обеспечивающее диспетчеризацию памяти, возможность организации многопроцессорных систем, обнаружение и коррекцию ошибок информации, введение периферийного арифметического процессора;

использование различных конструктивных исполнений микросхем: металлокерамические планарные корпуса, пластмассовые и металлокерамические корпуса типа ДИП;

переход на перспективную КМОП-технологию изготовления.

Высокие технические характеристики, совместимость по интерфейсу, системе команд и программному обеспечению с рядом микроЭВМ «Электроника 60», программная совместимость с рядом СМ ЭВМ (СМ-3/СМ-4) определяют широкую перспективу применения однокристальных процессоров серии К1801.

Статья поступила 16 октября 1984 г.

УДК 681.3—181.4

Ю. Д. Алексеев, А. В. Бойченко

ПЕРИФЕРИЙНЫЕ УСТРОЙСТВА мини- и микроЭВМ

Ориентация мини- и микроЭВМ на конкретного пользователя определяет, в основном, структуру и состав периферийных устройств (ПУ), к которым относятся внешние запоминающие устройства (ВЗУ) и устройства ввода-вывода информации (УВВ). В связи со значительным снижением стоимости машинного времени по сравнению с большими машинами информацию в мини- и микроЭВМ целесообразно вводить не с заранее подготовленного носителя (перфолента, перфокарта или магнитный носитель), а использовать диалоговый режим.

Основным устройством ввода информации в мини- и микроЭВМ становится дисплей с клавиатурой, а устройством вывода — печатающие устройства, обладающие большим диапазоном скоростей печати, возможностями графического вывода, цветной печати и т. д.

Для ввода и вывода графической информации в мини- и микроЭВМ применяются цифрователи (графоповторители), графопостроители, графические дисплеи, которые преобразуют графические изображения в последовательность кодированных сигналов для обработки в ЭВМ модели изображения с последующим выводом результатов в форме графического документа. Эти устройства отличаются друг от друга размерами вводимого и выводимого изображения, интеллектуальными и сервисными возможностями.

Проблемная ориентация вычислительных средств на базе мини- и микроЭВМ требует развития как УВВ, обеспечивающих общение пользователя с ЭВМ, так и различных классов ВЗУ, отличающихся емкостными, скоростными и эксплуатационными характеристиками.

В настоящем обзоре рассматриваются особенности и основные характеристики двух конструктивно и функционально наиболее развитых групп ПУ для мини- и микроЭВМ: ВЗУ на магнитных носителях и устройства ввода-вывода.

Внешние запоминающие устройства

ВЗУ делятся на два больших класса — *устройства с прямым доступом*, т. е. такие, в которых время доступа к информации мало зависит от ее расположения на магнитном носителе, и *устройства с последовательным доступом*, в которых эта зависимость оказывается значительной. К первым относятся различные виды накопителей на магнитных дисках, ко вторым — накопители на магнитных лентах. Основные характеристики ВЗУ — емкость, среднее время доступа к информации и скорость ее передачи в оперативную память процессора. По емкости и скорости обмена информацией накопители на магнитных дисках и накопители на магнитных лентах примерно соответствуют друг другу. По времени доступа накопители на дисках значительно лучше, но стоимость хранения единицы информации на ленте существенно ниже, чем на диске.

Накопители на магнитных дисках (НМД). Разновидности НМД отличаются друг от друга диаметром диска, материалом, из которого изготовлен диск, сменными или фиксированными дисками (или их комбинациями), подвижными или фиксированными головками записи-считывания, а также некоторыми особенностями конструктивного исполнения.

В соответствии с набором этих параметров можно выделить следующие типы НМД:

постоянные диски с фиксированными головками;
постоянные диски с подвижными головками;
со сменным пакетом дисков;
с комбинацией сменных и постоянных дисков;
типа Винчестер;
гибкие магнитные диски.

Постоянные диски с фиксированными головками. Постоянные диски жестко связаны с механизмом привода и не рассчитаны на смену в процессе работы. В дисках с фиксированными головками на каждую дорожку имеется отдельная головка записи-воспроизведения. В связи с тем, что в таких НМД время позиционирования головки над нужной дорожкой равно нулю, они имеют самые лучшие показатели по времени доступа. Число дорожек на одной поверхности в таких накопителях обычно составляет 64, 128 или 256. Число поверхностей от 1 до 16, причем наиболее типичны значения 2 и 4. Такое соотношение дорожек и поверхностей позволяет получать емкости от 1,2 до 10 Мбайт. Типичная скорость обмена — 0,5 Мбайт/с. Диаметр дисков 356 мм. Данный тип НМД не очень распространен. Если общее число моделей всех типов НМД, существующих в настоящее время на мировом рынке, принять за 100%, то данный тип НМД будет составлять 2...3%.

Постоянные диски с подвижными головками. Этот тип НМД отличается простотой конструкции в сочетании с достаточно хорошими емкостными и временными характеристиками. Диаметры дисков: 356 мм (4% всех моделей НМД), 200 мм (2%). В последнее время появилось несколько моделей накопителей с диаметром 130 мм (табл. 1).

Таблица 1

Диаметр диска, мм	Число рабочих поверхностей	Число дорожек на поверхность	Емкость, Мбайт	Среднее время позиционирования, мс	Скорость обмена, Мбайт/с
356	40	404...1616	12...675	25...50	0,9...1,2
200	2...5		10...165	30...65	0,9...1,2
130	4...6	306, 612	12...38	85	0,625

НМД со сменным пакетом дисков (17% всех моделей). Это достаточно традиционные устройства, большинство из которых имеют аналогами сменные пакеты фирмы Control Data 9876, 9877 (диаметр 356 мм, 5 рабочих поверхностей, 404 и 808 дорожек на поверхность) и IBM 3336-I, 3336-II (диаметр 356 мм, 19 рабочих поверхностей, 404 и 808 дорожек на поверхность). Емкости таких накопителей от 40 до 300 Мбайт. Среднее время позиционирования 28...30 мс, скорость обмена информацией 0,8...1,2 Мбайт/с. Эти НМД применяются в системах сбора и обработки больших

массивов данных с относительно малым временем доступа к ним.

Накопители с комбинацией сменных и постоянных дисков (как правило, один сменный диск диаметром 356 мм и несколько постоянных) позволяют сочетать преимущества постоянных и сменных дисков. Число рабочих поверхностей в комбинации, как правило, равно четырем, но может достигать 20. Постоянные диски по сравнению со сменными позволяют получать большую плотность дорожек, поэтому на сменном диске, как правило, число дорожек составляет 400, а на постоянном — 800. Аналогами наиболее массовых сменных кассет являются сменные диски в кассете фирмы IBM 2315 и 5440. Емкости таких накопителей 3...100 Мбайт, но наиболее характерны значения 6,4 и 12,8 Мбайт. Среднее время позиционирования 30...40 мс. Скорости обмена 0,195...1,2 Мбайт/с. Число моделей данного типа составляет примерно 9% от числа всех моделей НМД.

Накопители типа Винчестер или винчестерские накопители. Первая модель НМД (модель 3340), названная впоследствии винчестерским НМД, была выпущена фирмой IBM в 1973 году. Новшества, введенные в модель, известны как винчестерская технология.

Все винчестерские НМД характеризуются следующими особенностями:

— диски, головки чтения-записи и привод головок помещены в герметично закрытую конструкцию, в которой воздух постоянно циркулирует и фильтруется;

— зазор между головками чтения-записи и поверхностью диска значительно меньше, чем у обычных НМД (примерно 0,5 мкм);

— давление прижима головки составляет всего около 10 г по сравнению с 350 г в традиционных НМД;

— когда диск не вращается, головки лежат в специальной посадочной зоне на поверхности диска;

— поверхность диска покрыта специальной смазкой для предотвращения повреждения головок или дисков во время подъема и приземления;

— магнитно-оксидное покрытие диска тоньше, чем у других НМД.

Винчестерские НМД имеют преимущества перед традиционными. Герметизация значительно увеличивает надежность дисков и головок. Наработка на отказ для винчестерских НМД составляет от 8 до 12 тыс. ч по сравнению с 4...6 тыс. ч для обычных НМД.

Уменьшение зазора между головками и диском, а также малая толщина магнитного покрытия диска позволили значительно увеличить продольную и поперечную плотность записи. Так для винчестерских НМД диаметром 356 мм число дорожек на одной поверхности может достигать 1770.

В первом винчестерском накопителе IBM 3340 использовался *сменный модуль* с дисками и головками, а в настоящее время практически все эти НМД содержат *постоянные диски,* так как накопитель со сменным модулем оказывается значительно дороже НМД с соответствующим сменным пакетом.

В ранних моделях винчестерских НМД применялись диски диаметром 356 мм, однако в последнее время на мировом рынке появились модели с дисками 200 и 130 мм. В настоящее время соотношение между этими НМД следующее — число моделей с диаметром 356 мм составляет примерно 9% от числа всех моделей НМД на мировом рынке, с диаметром 200 и 130 мм — соответственно 16% и 9%. Основные характеристики винчестерских НМД приведены в табл. 2.

Таблица 2

Диаметр диска, мм	Число рабочих поверхностей	Число дорожек на поверхность	Емкость, Мбайт	Среднее время позиционирования, мс	Скорость обмена, Мбайт/с
356	2...40	32...1770	16,4...1300	16...54	0,8...1,2
200	1...8	185...1172	5,33...348	20...77	0,5...1,2
130	2...8	140...763	3...55	25...170	0,625...0,96

Характерная особенность винчестерских НМД — использование их вместе с поддерживающим устройством, представляющим собой накопитель (желательно дешевый) со сменным носителем для создания библиотек программ и данных. Наиболее распространенными поддерживающими устройствами для винчестерских НМД с дисками диаметром 200 мм являются накопители на гибких магнитных дисках (НГМД), с кассетной лентой шириной 6,35 мм и так называемые потоковые НМЛ с лентой шириной 12,7 мм.

Обычно рекомендуется использовать НГМД как поддерживающие для винчестерских дисков емкостью, меньшей 10 Мбайт. Копирование всего содержимого таких дисков требует восьми дискетт и занимает около 17 мин. Однако, если копировать не все содержимое диска, а только часть, необходимую для текущей работы, то достаточно НГМД емкостью 1 Мбайт.

Для винчестерских НМД емкостью 15...40 Мбайт в качестве поддержки рекомендуются 6,35 мм *кассетные НМЛ*. Содержимое диска емкостью 20 Мбайт может быть скопировано на ленту примерно за 70 с. *Потоковые НМЛ* обмениваются данными с винчестерскими НМД непрерывным потоком без старт-стопа между блоками данных, так как у них нет межблочных промежутков. Они сконструированы специально для загрузки и копирования целых файлов на стандартную ленту шириной 12,7 мм и сто́ят дешевле аналогичных старт-стопных НМЛ. Такие НМЛ рекомендуются как поддерживающие устройства для винчестерских НМД емкостью от 40 Мбайт.

Накопители на гибких магнитных дисках (НГМД). Носителем в таких устройствах является гибкий майларовый диск (дискетта), покрытый с одной или двух сторон магнитным слоем и помещенный в специальный конверт.

В зависимости от типа носителей можно выделить НГМД с мягким и жестким секторированием дискетты, с мини- и микродискеттой. Эти типы НГМД взаимозаменяемы с точки зрения носителей. В НГМД первых двух типов используются дискетты диаметром 200 мм

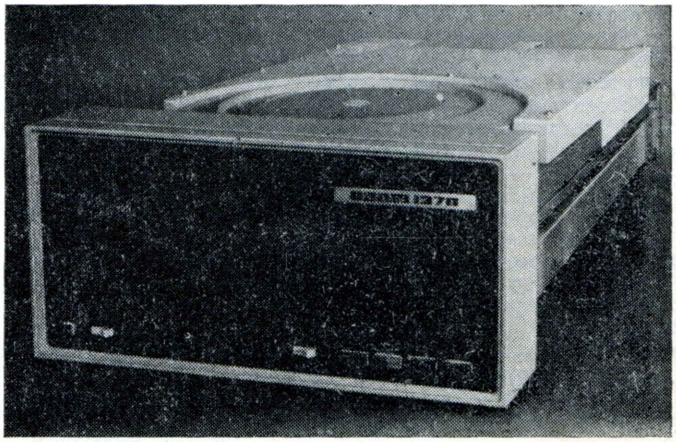


Рис. 1. НМД ИЗОТ 1370 (СМ 5400) (НРБ)

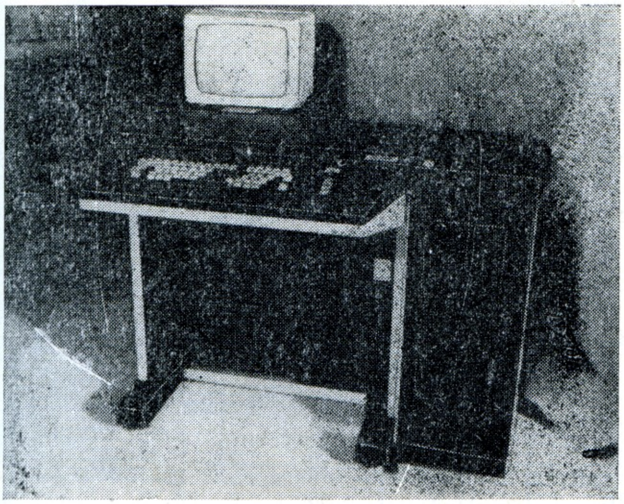


Рис. 2. Система подготовки информации на НГМД СМ 6390

(примерно 15% всех типов НМД), толщиной примерно 0,8 мм и 38 мм отверстием в центре.

В НГМД с *мягким секторированием* (формат устройства IBM 3740) единственное индексное отверстие размещено в центре для идентификации начала дорожки. Разметка каждой дорожки на 26 секторов по 128 байт каждый (всего 77 дорожек) производится электронным способом. Четыре дорожки резервируются как запасные и индексные, что обеспечивает форматированную емкость 0,243 Кбайт или неформатированную — около 0,4014 Мбайт.

НГМД с *жестким секторированием* используют 32 отверстия в дополнение к индексному для разбиения дорожек на секторы. При этом 32 сектора по 128 байт каждый на 77 дорожках дают общую емкость в 0,315 Мбайт.

Мини-дискетты (диаметром 130 мм) (примерно 17% всех моделей НМД) появились на мировом рынке в 1976 году. Они также подразделяются на дискетты с мягким и жестким секторированием. При мягком секторировании используется единственное индексное отверстие, при жестком — 16 секторных отверстий. Сектора содержат 128 или 256 байт при записи на 35 дорожках.

Необходимо отметить, что для дискетт с диаметрами 200 мм и 130 мм наиболее распространено мягкое секторирование. Основные характеристики НГМД с диаметрами дискетты 200 мм и 130 мм приведены в табл. 3.

Таблица 3

Диаметр дискетты, мм	Емкость при записи на одну поверхность, Мбайт	Время смены дорожек, мс	Число поверхностей
130	0,125...1,6	2...30	1 (46%), 2 (54%)
200	0,3...1,6	1,2...23	1 (54%), 2 (46%)

В последнее время появилось несколько моделей НГМД фирм Sony (Япония) и Shugart Ass (США) с так называемыми микродискеттами диаметром 89 мм. Они имеют емкость 437,5 Кбайт и скорость передачи данных 62,5 Кбайт. Малые габариты (примерно 5×10×12 см) и масса этих устройств (примерно 700—800 г) делают их весьма перспективными для различных микроЭВМ.

В табл. 4 приведены основные характеристики некоторых НМД, имеющихся на мировом рынке в настоящее время, а также выпускаемых странами—членами СЭВ.

Накопители на магнитной ленте (НМЛ) представляют собой устройства с последовательным доступом к памяти; большой емкостью и временем доступа; меньшими, по сравнению с НМД, скоростями обмена информацией; малыми габаритами и самой низкой стоимостью хранения информации. В настоящее время наибольшее распространение получили НМЛ со стандартной бобинной магнитной лентой и НМЛ со стандартными мини-кассетами (КНМЛ).

Бобинные НМЛ используют магнитную ленту шириной 12,7 мм. Информация записывается на девяти дорожках, т. е. поперек ленты записывается 1 байт и одна дорожка служит контрольной. Лента наматывается на бобины диаметром 267 мм (длина 730 м, 88% всех моделей), 216 мм (365 м, 7%) или 178 мм (183 м, 5%).

Емкость НМЛ зависит от длины магнитной ленты и продольной плотности записи. В настоящее время применяются *три способа записи* — *без возврата к нулю*, дающий плотность записи до 32 бит/мм (38%); *фазовая модуляция*, обеспечивающий плотность записи 63 бит/мм (46%); *групповое кодирование* с плотностью 246 бит/мм (16%). При указанной плотности записи и длине ленты достигается емкость НМЛ от 12 до 180 Мбайт.

Следующая важная характеристика НМЛ — скорость обмена информацией — определяется плотностью записи и скоростью движения ленты относительно головки. Существенно, что увеличение скорости движения ленты повышает требования к буферированию ленты и приводит к необходимости использовать довольно сложные вакуумные колонки вместо рычажных буферов, влечет за собой значительное увеличение габаритов устройства. Определившийся в настоящее время на мировом рынке диапазон скоростей составляет 0,15—3,5 м/с. Наиболее типичными являются значения 0,3 м/с (5%), 0,635 м/с (13%), 1 м/с (20%), 2 м/с (20%) и 3 м/с (25%). Такой диапазон скоростей движения ленты при существующих плотностях записи обеспечивает скорость обмена информацией от 10 до 780 Кбайт/с.

В развитии НМЛ наметились две тенденции — постоянное увеличение числа моделей с записью способами фазовой модуляции и группового кодирования и появление НМЛ, выполненных по так называемой потоковой технологии (о потоковых НМЛ говорилось в разделе винчестерских НМЛ). Доля потоковых НМЛ среди всех моделей НМЛ составляет ~ 22%.

Кассетные НМЛ являются наиболее компактными и дешевыми устройствами из всей номенклатуры ВЗУ. Это определило их широкое применение в мини- и микроЭВМ в качестве устройств внешней памяти и перспективных устройств ввода-вывода.

В последнее время наиболее широкое распространение получили два вида кассет для КНМЛ — кассеты стандарта 3М (52%) и филлипс кассеты (35%). В кассетах 3М используется лента длиной от 45 до 180 м и шириной 6,35 мм. Типовое число дорожек равно 2 и 4. При плотностях записи 32 бит/мм (5%), 63 бит/мм (65%) и 246—390 бит/мм (30%), указанных длинах ленты и числе дорожек достигается емкость записи от 0,168 до 60 Мбайт. Скорости движения ленты от 25 до 220 см/с.

В филлипс кассетах применяется лента шириной 3,81 мм и длиной от 90 до 120 м. Число дорожек составляет 1,2 или 4 (наиболее типична запись по двум дорожкам). Плотность записи от 6,3 до 390 бит/мм. Наиболее типичное значение плотности — 63 бит/мм. Значения емкости при указанных параметрах — от 72 Кбайт до 25 Мбайт.

Таблица 4

Тип	Шифр устройства, страна-изготовитель	Емкость, Мбайт	Среднее время доступа, мс	Скорость передачи, Мбайт/с
Постоянные диски с фиксированными головками	Dataflux Model ДС-111, США	2,495	8,5	0,512
	СМ 5500-03 (МД 1000-С), ВНР	1,024	10	1,35
Постоянные диски с подвижными головками	Century Data. Systems Marksman M20, США	20,16	77,5	0,96
	Control Data 9410, США	40	51,3	6,45
	Dicom Industries Model 604, США	14,5	97,1	0,889
НМД со сменным пакетом дисков	DEC RM02, США	80	42,5	0,806
	СМ 5412, НРБ	80	45	0,8/1,2
Накопители с комбинацией сменных и постоянных дисков	Control Data 9427 H, США	6,3	47,5	0,312
	HP 7906 M, США	25,7	33,3	0,937
	СМ 5400 (ИЗОТ 1370), НРБ (рис. 1)	6,3	60	0,312
	СМ 5410 (ИЗОТ 5410 С), НРБ	11,5	50	0,18
Накопители типа Винчестер	BASF 6173, ФРГ	40,97	35,3	0,8
	Diva DD-56W, США	82,9	38,3	1,209
	СМ 5411, ГДР	35	52,5	0,531
Накопители на гибких магнитных дисках	Control Data 9404B, США	0,4/0,8	340	31 Кбайт/с 62 Кбайт/с
	DEC RX02, США	0,5/1	250	62 Кбайт/с
	СМ 5602 (PL×45), ПНР	0,8	500	31 Кбайт/с
	СМ 5601 (МФ-3200), ВНР	0,4	500	31 Кбайт/с
	ЕС 5074, НРБ (рис. 2)	0,4	500	31 Кбайт/с

Среди перспективных КНМЛ существует тенденция к увеличению числа потоковых моделей. Для КНМЛ с кассетой 3М доля таких моделей составляет 20%, а для филлипс кассет — 18%. Значения характеристик наиболее распространенных НМЛ и КНМЛ приведены в табл. 5.

Устройства ввода-вывода информации

В качестве УВВ мини- и микроЭВМ в настоящее время используются в основном перфоленточные устройства, дисплеи и печатающие устройства с клавиатурой. Скоростные характеристики этих устройств, как правило, соответствуют средним показателям аналогичных устройств для средних и больших ЭВМ. В последнее время наблюдается тенденция применения в качестве УВВ НГМД и КНМЛ, которые заменяют устройства ввода с перфоленты и перфокарты.

Перфоленточные устройства ввода-вывода. К основным характеристикам перфоленточных УВВ относятся: скорость работы; возможность работать в непрерывном и в старт-стопном режимах с 5—8 дорожечными перфолентами; способ считывания информации; способность работать с автономными приемопередающими устройствами, на которых размещаются кассеты для перфоленты; надежность.

В качестве носителя информации в этих УВВ используются бумажные, майларовые и пластмассовые перфоленты. По способу считывания различают фотоэлектрические, механические (электроконтактные) с воздушным принципом считывания и емкостные. Наиболее распространены *фотоэлектрические* УВВ со скоростью считывания 1000—1500 строк/с. Для *механических* скорость считывания составляет несколько сотен строк в секунду; с *воздушным принципом считывания* — 300 строк/с; *емкостных* — 1000 строк/с.

Перфоленточные УВВ выполняются в виде отдельных устройств, которые могут встраиваться в основной

Таблица 5

Тип	Шифр устройства, страна-изготовитель	Емкость, Мбайт	Скорость обмена, Кбайт/с
НМЛ	Control Data 9214X, США	40	72
	HP 7979 B, США	20	36
	СМ 5300 (ИЗОТ 5004Е), НРБ	11,5	10
	СМ 5309, НРБ	40	72
	СМ 5305, СССР	46	64/126
КНМЛ (филлипс кассета)	Memodyne 764, США	0,625	4
	TEAC MT2-02, США	0,5	1,5
	СМ 5205 (РК-1), ПНР (рис. 3)	0,72	0,5
	СМ 5206 (К 5200), ГДР	0,72	0,76/1,52
КНМЛ (кассета 3М)	Digi-Data Series 70, США	17,3/30	24/30
	Kennedy 6455, США	17,3/23	24

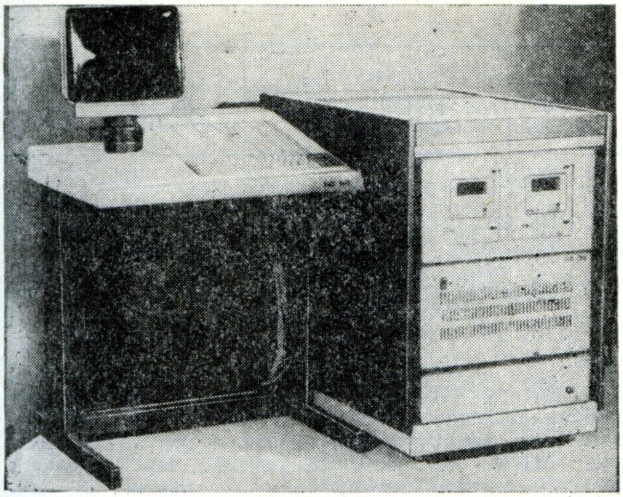


Рис. 3. УПД на КНМЛ РК-1 (ПНР)

шкаф машины. В современных встраиваемых устройствах вместо приемной и подающей бобин имеются карманы для использования фальцованной перфоленты, что значительно уменьшает их габариты.

Для мини-ЭВМ применяются комбинированные перфоленточные УВВ (в большинстве случаев встраиваемые). Они представляют собой как правило среднескоростной считыватель (300 ... 500 строк/с) и перфоратор со скоростью вывода 50 ... 75 строк/с, объединенные в одном устройстве. Основные характеристики перфоленточных УВВ приведены в табл. 6.

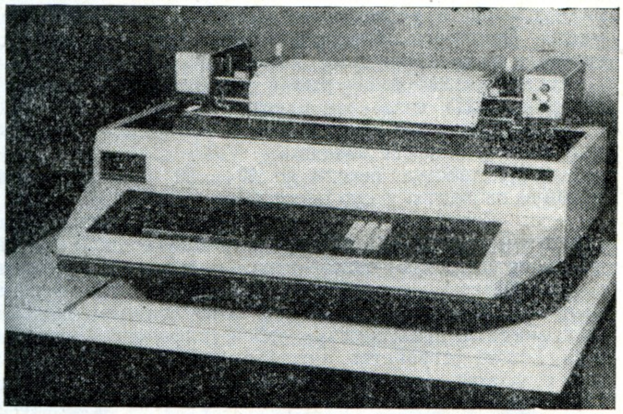


Рис. 4. АЦПУ ДЗМ-180 с клавиатурой (ПНР)

Перфоленточные УВВ входят в состав практически всех УВК на основе мини-ЭВМ, причем в базовых комплексах применяется встраиваемая модель комбинированного УВВ типа СМ 6204. Быстродействующие настольные перфоленточные УВВ типа СМ 6209 менее распространены.

Основная тенденция развития УВВ малых ЭВМ — постепенная замена перфоленточных УВВ на более перспективные и удобные в эксплуатации КНМЛ и НГМД.

Печатающие устройства. Алфавитно-цифровые печатающие устройства (АЦПУ) — наиболее развитая группа устройств вывода информации ЭВМ, реализующих различные методы печати и имеющих большое число конструктивных решений. АЦПУ можно разделить по *способу печати* на ударные и безударные, по *способу формирования* символов — на знакопечатающие и знакосинтезирующие, в том числе матричные, по *способу формирования* строки — на устройства параллельной и последовательной печати.

Широкий диапазон технических и эксплуатационных характеристик, разнообразие типов и видов позволяет использовать АЦПУ в УВК на базе мини- и микроЭВМ практически для всех областей применения. Децентрализованная обработка информации, ориентация на применение дисплеев для оперативного ввода информации и ВЗУ на НГМД и КНМЛ, характерные для систем на базе микроЭВМ, определяют дальнейшее развитие и широкое использование АЦПУ в качестве устройства визуального вывода информации, в том числе с экрана дисплея.

Наибольшее распространение в УВК на базе мини- и микроЭВМ в настоящее время получили *матричные ударные* АЦПУ последовательного типа (59%).

При сравнительно невысоком быстродействии (100 ... 180 знаков/с) они обладают небольшими размерами, возможностью изменения набора печатных символов в широких пределах и сравнительно низкой стоимостью. Эти устройства часто объединяются с клавиатурой и применяются в качестве устройства связи оператора с машиной (рис. 4).

В отдельных матричных АЦПУ реализована возможность вывода простейшей графической информации. Увеличение растра матрицы позволяет использовать эти АЦПУ для печати коммерческой информации. Для вывода информации с цветных дисплеев в ряде матричных АЦПУ (10%) применяется многоцветная печать.

Исходя из характеристик и особенностей использования матричных АЦПУ в мини- и микроЭВМ, и мож-

Таблица 6

Тип	Шифр устройства, страна-изготовитель	Основные характеристики		
		Скорость, строк/с	Число дорожек	Возможность встраивания
Устройства ввода с ПЛ	СМ 6203 (МР-301), ВНР	500	5,8	Имеется
	СМ 6205 (СТ-2030), ПНР	300	5,8	Нет
	СМ 6216 (СТ-2100/2200),) ПНР	1000/3000	5,8	Нет
	СМ 6208 (Консул 337.2), ЧССР	300	8	Имеется
	СМ 6209 (ФС-1503), ЧССР	1500	8	Нет
Устройства вывода на ПЛ	СМ 6222 (ДТ-105С), ПНР	50	5,8	Имеется
	СМ 6227 (МП-51), ВНР	50	5,8	»
Комбинированные УВВ	СМ 6200 (МПР-51/301), ВНР	50/500	5,8	»
	СМ 6204 (СПТП-3), ПНР	50/500	5,8	»
	СМ 6211 (Роборон К 6200), ГДР	50/300	5,8	»

Таблица 7

Скорость печати, знаков/с	Доля, %	Размер матрицы	Доля, %	Набор символов	Доля, %	Ширина строки, знаков	Доля, %	Наличие графики	Доля, %
Не более 80	13	7×7(8)	7	64	5	Не более 80	15	Нет	35
100...120	17	5×7—9	12	96	66	132	40	Блок графики	17
140...180	3[illegible]	7×9	34	Не более 128	18	136	16	72×72	21
200	18	9×9	31	Не более 192	9	198	5	72×144	11
220...350	10	9×11—18	10	Более 255	2	до 220	12	144×160	13
Более 400	6	18×20—58	6			более 220	12	240×360	3

но условно разделить на несколько групп:

— простые (со скоростью печати до 100 знаков/с);

— с повышенной интеллектуальностью (с возможностью вывода простейшей графической информации);

— скоростные (до 200 знаков/с);

— с высокой интеллектуальностью (вывод графики), с увеличенным (уплотненным) растром для повышения качества печати;

— высокоскоростные (более 200 знаков/с).

Доля матричных АЦПУ, наиболее часто применяемых при комплектовании систем, приведена в табл. 7, в соответствии с их основными параметрами.

Следует отметить, что высокоскоростные матричные АЦПУ из-за быстрого износа некоторых элементов и необходимости проведения частой профилактики не могут использоваться для вывода больших объемов информации. Здесь целесообразно применять АЦПУ параллельного типа со скоростью печати 200...1600 строк/мин и безударные быстродействующие АЦПУ.

Наиболее перспективными для применения в мини- и микроЭВМ являются АЦПУ с ленточным шрифтоносителем, имеющие меньшие габариты при одинаковой скорости печати. В некоторых среднескоростных АЦПУ скорость печати можно повысить за счет использования сокращенного набора символов или уменьшения числа позиций в строке. Основные характеристики современных АЦПУ параллельного типа и их процентное соотношение приведены в табл. 8.

Таблица 8

Скорость печати, строк/мин.	Доля, %	Набор символов	Доля, %	Ширина строки, знаков	Доля, %
Не более 400	20	48	20	132	89
„ 700	24	64	41	136	3
„ 1000	23	96	36	198	5
„ 1600	25	128	3		
Более 1800	8				

В группу АЦПУ последовательного типа входят знакопечатающие устройства, позволяющие выпечатывать знаки замкнутой конфигурации. Эти низкоскоростные АЦПУ (скорость печати 20...60 знаков/с) с шрифтоносителем типа «ромашка» составляют 10% всех применяемых АЦПУ (табл. 9).

Знакопечатающие АЦПУ последовательного типа первоначально предназначались для использования вместо телетайпа. Сейчас они успешно конкурируют с матричными и заменяют их при печати на низких скоростях, обеспечивая более высокое качество печати.

По сравнению с матричными АЦПУ устройства этого типа обладают более простым (механическим) способом замены шрифтоносителя. Срок службы «ромашки» — 4 млн. ударов для каждого лепестка. Возможность изменения расстояния между строками и знаками по горизонтали и по вертикали позволяет в ряде устройств реализовать вывод графической информации. Основным параметром этих устройств является скорость печати, которая определяет их стоимость.

На международном рынке можно выделить три категории АЦПУ типа «ромашка»:

— со скоростью печати до 25 знаков/с, имеющие простые технические решения;

— со скоростью печати до 40...50 знаков/с, составляющие основную долю этого типа устройств;

— со скоростью печати до 80 знаков/с, выпускаемые отдельными производителями для определенных целей, имеющие сложную конструкцию и высокую стоимость.

Безударные печатающие устройства, применяемые главным образом, в системах автоматизации научного эксперимента, представляют собой графическое знакосинтезирующее устройство, позволяющее осуществлять вывод печатной (до 100 символов/с) и графической (до 100 мм/с) информации. Эти устройства составляют довольно большую группу печатающих устройств (13% существующих АЦПУ), реализующих термографический, электростатический, электротермический и другие принципы регистрации печатаемых символов на носителе, которым является, в основном, специальная бумага. Скорость печати безударных АЦПУ достигает более 2000 строк/мин. Однако эти устройства пока не нашли достаточно широкого применения из-за низкой надежности, сложности в эксплуатации и высокой стоимости специальной бумаги.

Таблица 9

Скорость печати, знаков/с	Доля, %	Набор символов, знаков	Доля, %	Ширина строки, знаков	Доля, %	Наличие графики	Доля, %
Не более 25	16	88	22	132	40	Нет	52
„ 40	39	96	58	158	27	Есть	18
„ 50	32	127	20	198	24	48×120	20
Более 55	13			Более 225	9	96×120	10

Таблица 10

Тип устройства	Шифр устройства, страна-изготовитель	Основные характеристики		
		Скорость печати	Длина строки, символов	Набор символов
АЦПУ параллельного типа	РСД-9322 (СМ 6322), СРР	200 строк/мин	132	64, 96
	ВТ-24112 (СМ 6316), ВНР	253 строк/мин	80	96, 64
	Teletype T/300, США	300/200 строк/мин	132	64, 96, 128
	General Electric, GE 340, США	231—425 строк/мин	132	48, 64, 96
	СМ 6315.01, СССР	500 строк/мин	132	96
	ВТ-27065 (СМ 6313), ВНР	660 строк/мин	136	96
	Dataproducts Corp. B-600, США	650 строк/мин	132, 136	48, 64, 96
	ВТ-25122 (СМ 6306), ВНР	900 строк/мин	132	96
Матричные АЦПУ	DARO-1156 (СМ 6301), ГДР (рис. 5)	100 знаков/с	132	96
	А-621-4 (СМ 6308), СССР	100 знаков/с	128	96
	DECLA-50, США	100 знаков/с	40—132	128
	Dataproducts Corp. M-100, США	140 знаков/с	132	128
	Facit 4528, Швеция	165 знаков/с	80/136	96
	ДЗМ-180 (СМ 6302), ПНР	180 знаков/с	132, 158	128
	Tally MT-1802, США	200/50 знаков/с	132, 218	96
	Centronics 351, США	200 знаков/с	до 132	96
	Роботрон 1157 (СМ 6309), ГДР	80, 160, 320 знаков/с	132	96
АЦПУ типа «ромашка»	Роботрон 1152 (СМ 6317), ГДР	45 знаков/с	132	96
	Olympia ESW 3000, ФРГ	50 знаков/с	150—225	100
	Fujitsu SP 320, Япония	48 знаков/с	136	96/127

Особый интерес представляют *струйные* АЦПУ, обладающие рядом достоинств: бесшумная работа при достаточно высокой скорости печати (до 100 знаков/с); высокая разрешающая способность (до 1000 точек в знаке); возможность печати на различных материалах (бумага, ткань и т. д.); возможность многоцветной печати (до тысячи различных цветов и оттенков), что особенно важно при использовании этих АЦПУ совместно с многоцветными дисплеями. Уже в настоящее время имеются примеры успешной замены матричных АЦПУ струйными (фирма «Сименс», ФРГ) для ряда применений.

Основные характеристики различных типов печатающих устройств мини- и микроЭВМ приведены в табл. 10. В целом группу АЦПУ мини- и микроЭВМ следует охарактеризовать как функционально полный набор устройств визуального вывода информации для различных сфер применения. Развитие такого вида ПУ направлено на увеличение скорости печати и функциональных возможностей при сохранении или уменьшении габаритов. Это важно при повышении производительности УВК и использовании микропроцессоров. Характерной тенденцией развития этого вида ПУ является возможность и целесообразность разработки нескольких типов устройств для удовлетворения потребностей каждой из областей применения вместо создания сложных универсальных устройств. Одним из основных направлений развития печатающих устройств для мини- и микроЭВМ в целом является повышение эксплуатационных характеристик (надежности, простоты обслуживания, унификации для различных применений и т. д.) при одновременном снижении стоимости.

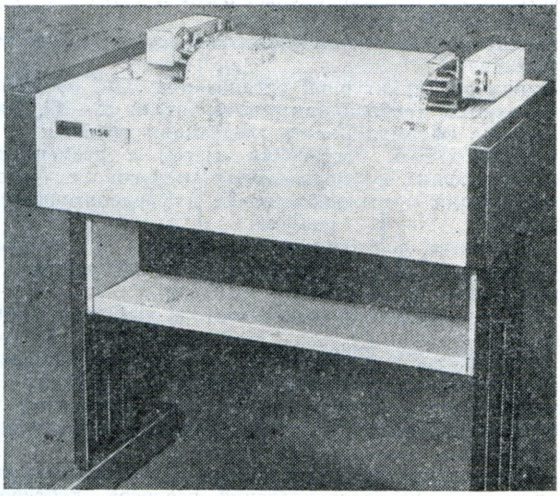


Рис. 5. Мозаичное АЦПУ Daro 1156 (ГДР)

Рассмотренные группы ПУ отвечают требованиям функциональной полноты систем на базе мини- и микроЭВМ и обеспечивают возможность построения вычислительных комплексов для основных категорий систем управления и обработки информации.

ЛИТЕРАТУРА

Отраслевой каталог: ГСП. Система малых ЭВМ. Общее описание /ЦНИИТЭИприборостроения. — М., 1983.

Каталог: Система малых электронных вычислительных машин. Разработанные и испытанные технические и программные средства /В/О «Союзагранприбор». — М., 1982.

Алексеев Ю. Д., Мячев А. А. Периферийные устройства СМ ЭВМ. Обзорная информация. — ТС-2, Средства ВТ и оргтехники. — М.: ЦНИИТЭИприборостроения. Вып. 1, 1980.

Савета Н. Н. Печатающие устройства. — М.: Машиностроение, 1977.

Рыжков В. А., Сергеев Н. П., Раков Б. М. Внешние запоминающие устройства на магнитном носителе. — М.: Энергия, 1978.

Альянах И. Н. Внешние запоминающие устройства ЕС ЭВМ. — М.: Советское радио, 1979.

Статья поступила после доработки 19 марта 1984 г.

УДК 681.327.8(3.82.7)+681.324.—181.4

Б. Н. Виноградов, В. А. Шахнов

РАСПРЕДЕЛЕННЫЕ МИКРОПРОЦЕССОРНЫЕ СИСТЕМЫ И ЛОКАЛЬНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ ЭВМ

Развитие и широкое распространение микропроцессорных средств вычислительной техники привело к необходимости объединения этих средств в системы, реализующие распределенную обработку данных — *распределенные микропроцессорные системы,* предназначенные для автоматизации физического и интеллектуального труда. К ним относятся системы «электронной почты», информационно-вычислительные, справочные и управляющие, системы для «безбумажного делопроизводства», системы автоматизированного проектирования (САПР), гибкие перепрограммируемые автоматизированные производства (ГАП) и т. д. [1—5].

При организации *распределенных вычислительных систем* (ВС) учитываются отдельные группы средств вычислительной техники: большие, средние, мини- и микроЭВМ, обладающие спецификой применения для отдельных задач пользователя. Особое место в создании вычислительных средств и систем занимают *локальные вычислительные сети ЭВМ (ЛВС),* объединяющие достижения вычислительной техники, микроэлектроники и техники связи. Техника ЛВС создает условия для построения на унифицированной основе наращиваемых многомашинных распределенных ВС с модульной развиваемой и адаптивной структурой, подключением функционально-ориентированных, специализированных, универсальных и высокопроизводительных средств сбора, обработки, хранения и ввода-вывода информации.

Сетевые принципы организации вычислительных систем могут наилучшим образом удовлетворить растущие потребности пользователей, предоставляя новые функциональные возможности:

доступность и многофункциональность;

высокую степень готовности к работе, надежность и живучесть;

высокую реактивность и динамичность;

наращиваемость;

Таблица 1

Архитектурная модель и аппаратно-программное обеспечение ЛВС

Номер уровня	Архитектурный уровень	Технические средства	Программное обеспечение
7	Прикладной	Пользовательская станция	взаимодействия процессов
6	Представительный		
5	Сеансовый		
4	Транспортный	Транспортная станция или коммутационный процессор (канал передачи), физическая среда	сетевого доступа
3	Сетевой		
2	Канальный		
1	Физический		

сопрягаемость с неоднородными вычислительными средствами и способность взаимного обмена данными.

ЛВС составляют основу структурной организации распределенных ВС, т. е. *включают необходимую среду обработки, средства передачи данных и процедуры взаимодействия процессов.* Архитектурная организация распределенных ВС базируется на архитектуре ЛВС, соответствующей семиуровневой стандартизированной модели ИСО [6—8] (табл. 1).

Реализации распределенных ВС предшествуют следующие этапы:

— выбор основных концепций построения ЛВС и системы передачи данных (СПД);

— разработка интерфейсов и отладка протоколов передачи сообщений в СПД и в подключаемых вычислительных средствах;

— формирование принципов построения базового комплекта аппаратных средств на основе микропроцессорных наборов и специализированных БИС;

— определение принципов архитектурной и функциональной организации процессов распределенной обработки;

— создание операционных систем, системного и прикладного программного обеспечения.

ЛВС: классификация, требования к структуре

ЛВС по назначению можно разделить на следующие четыре группы:

связывающие большие и супер-ЭВМ;

реализующие информационно-справочные системы, САПР и т. п.;

используемые в АСУ, с подключением неоднородных вычислительных средств с развитой сетью терминальных устройств и персональных ЭВМ и образующих локальную или региональную сеть;

объединяющие технологическое и измерительное оборудование.

Принципы структурной организации ЛВС во многом определяются требованиями потребителя [6—10], которые можно сформулировать следующим образом:

— возможность построения систем с размещением обрабатывающих и управляющих ресурсов в местах возникновения и вывода данных и управляющих воздействий;

— взаимная доступность технических и программных ресурсов ЭВМ, входящих в общую систему и подключенных к СПД;

— распределенность подготовки и хранения данных и программ;

— возможность программной настройки вычислительных ресурсов сети на распределенную обработку отдельных задач с распараллеливанием алгоритмов обработки для повышения интегральной производительности ВС;

— обеспечение отказоустойчивости системы за счет программного перераспределения ресурсов, дублирования и параллельной обработки с сопоставлением результатов.

Центральным звеном ЛВС, определяющим структуру и задающим динамичность, возможность взаимного функционирования технических ресурсов в реальном масштабе времени и надежность, является СПД, выполняющая задачи цифровой передачи данных между взаимодействующими парами ЭВМ с обеспечением гибкости и множественности связей, а также необходимой скорости передачи данных. В процессе проектирования СПД решаются задачи выбора топологии и средств построения каналов передачи.

Типы СПД и их характерные особенности

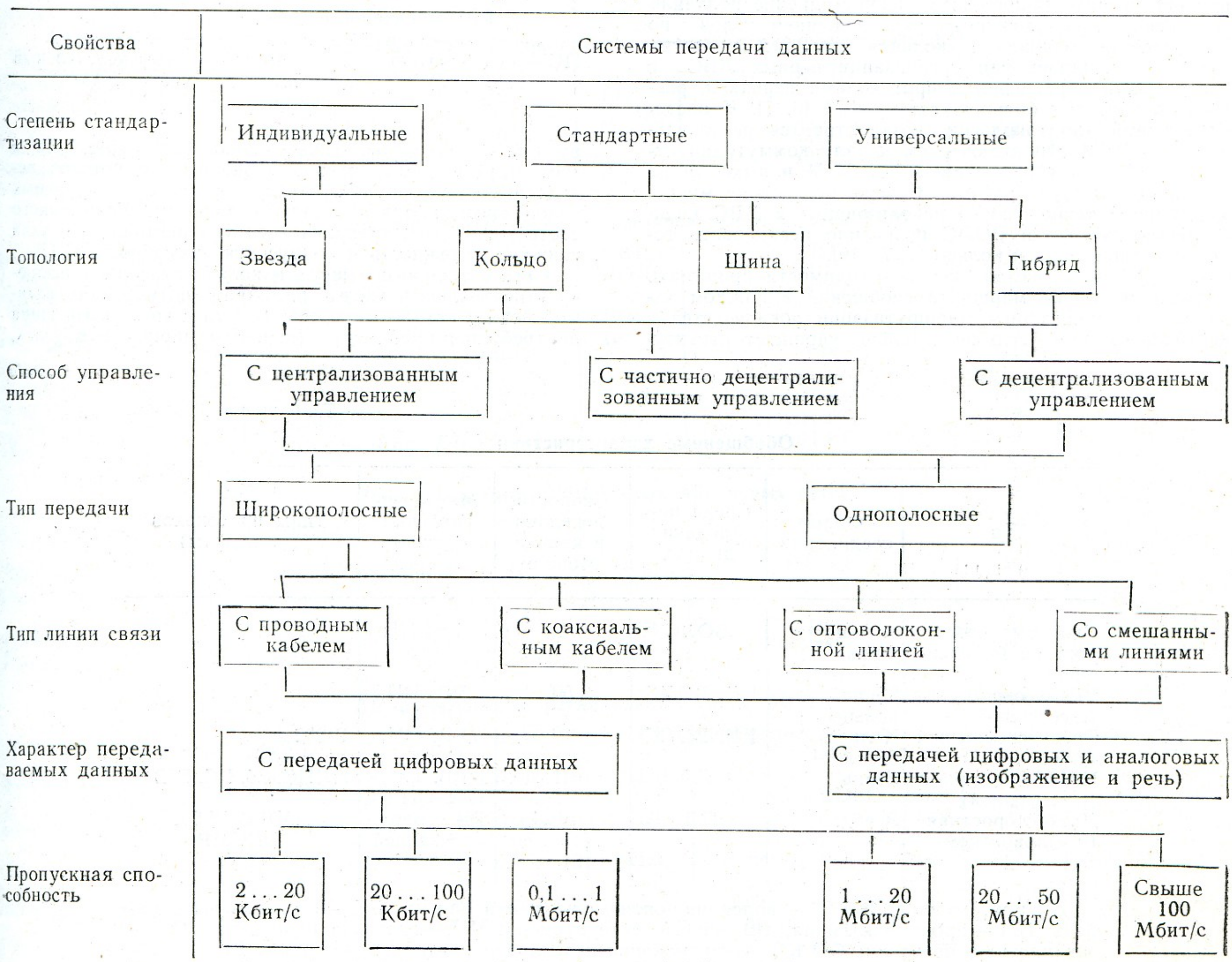
СПД характеризуются различными свойствами. В табл. 2 приведена общая классификация СПД локальных сетей ЭВМ [11]. Близость расположения вычислительных средств ЛВС дает возможность использовать в качестве среды передачи данных не только проводные линии, но коаксиальные кабели и волоконно-оптические линии связи (ВОЛС). Это позволяет увеличить скорость передачи с единиц Кбит/с до десятков Мбит/с и организовать «одновременную» работу на единый канал связи десятков и сотен ЭВМ, и тем обеспечить эффективность использования оборудования новых средств передачи данных [1, 7, 8, 12]. С помощью проводных линий осуществляется передача данных со скоростью десятков и сотен Кбит/с, коаксиальных кабелей — до десятков Мбит/с, ВОЛС — до сотен Мбит/с [1, 4, 8, 10] (рис. 1).

В организации ЛВС существенную роль играют выбор типа канала передачи данных и его топология, метод адресации и способ организации управления передачей. Возрастающая скорость обработки и преобразования данных техническими средствами распределенной ВС требует повышения пропускной способности каналов связи. Увеличение производительности и пропускной способности канала пропорционально росту потока данных между подключенными к нему средствами, производительности этих средств и их количеству.

При построении ЛВС находят применение СПД, отличающиеся методами управления доступом к среде, протоколами передачи и скоростными свойствами — низкоскоростные, среднескоростные, высоко- и сверхвысокоскоростные. Низкоскоростные СПД обеспечивают передачу в канале со скоростью десятков — сотен Кбит/с, среднескоростные — от 1 до 10 Мбит/с, высокоскоростные — до 50 Мбит/с, сверхвысокоскорост-

Таблица 2

Общая классификация СПД локальных сетей ЭВМ



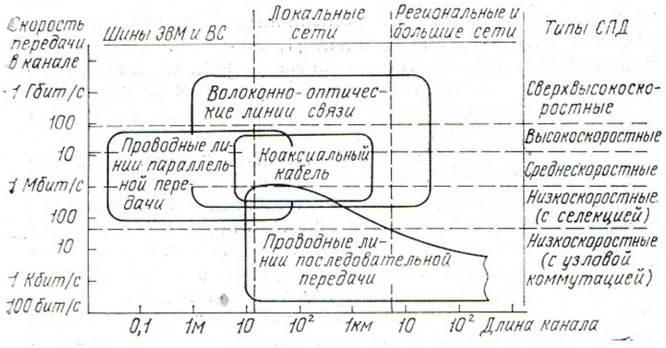


Рис. 1. Изменение свойств физической среды и пропускной способности каналов СПД в вычислительных системах и сетях ЭВМ

ные — свыше 50 Мбит/с до гигабитного диапазона (табл. 3). Широкое распространение в технике ЛВС получили наиболее доступные СПД на основе коаксиального кабеля, однако последнее время ведущее положение начинают занимать СПД с использованием ВОЛС, обладающие высокой пропускной способностью, помехозащищенностью и имитостойкостью [1, 7, 8, 13, 14].

Конфигурация каналов передачи СПД образует топологию ЛВС. Из известных топологических принципов организации наибольшее распространение получили топологии с упорядоченным расположением узлов по типу «звезда», «шина» и «кольцо» (рис. 2). Топология звезда применялась при организации первых ЛВС с процедурами управления и протоколами передачи, разработанными для больших сетей ЭВМ [6, 7]. В настоящее время по топологии звезда строятся региональные сети ЭВМ с подключением к узлу коммутации отдельных ЛВС через шлюзовые ЭВМ и выходом на большую сеть (рис. 2, *в*). Совершенствование микроэлектронной технологии и использование в ЛВС коаксиальных кабелей и ВОЛС позволило реализовать топологии типа шина и кольцо [1, 7—10].

В СПД типа звезда нет необходимости принимать решение о выборе маршрута сообщений в каждом узле, поскольку алгоритм маршрутизации локализуется в центральном узле. Это значительно упрощает структуру отдельных узлов. Такая топология выбирается в соответствии со структурой управления и потоков передачи сообщений, т. е. когда множество вторичных узлов связываются с одним первичным. Если такого соответствия нет, то наличие одного центрального узла снижает надежность и пропускную способность СПД.

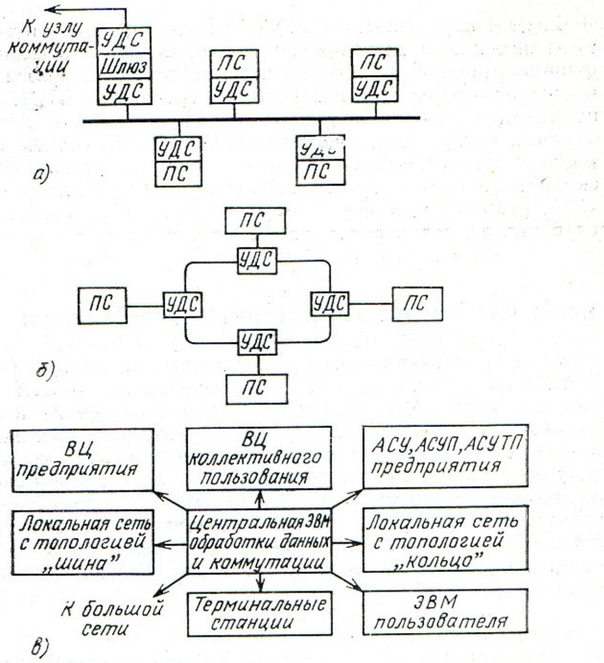


Рис. 2. Основные типы топологий ЛВС:
а — шина, *б* — кольцо, *в* — звезда
ПС — пользовательская система; ВЦ — вычислительный центр; УДС — устройство доступа к среде

Сети с топологией шина и кольцо позволяют исключить центральный узел и расширить возможность взаимодействия остальных узлов. В сети с топологией шина не требуется принимать решение о маршруте в узлах,

Таблица 3

Обобщенные характеристики СПД

Тип	Форма соединения	Среда передачи данных	Скорость передачи в канале, Мбит/с	Метод управления доступом к среде	Типы протоколов (стандарты)
Сверхвысокоскоростная (с селекцией)	Кольцо шина	ВОЛС	100 и выше	РВ, ПП	
Высокоскоростная (с селекцией)	Шина, кольцо	КК, ВОЛС	50	РВ, ПП, ОП-КС	
Среднескоростная (с селекцией)	Кольцо, шина	КК, ВОЛС	10	РВ, ПП, ОП-КС	Р-802 ЕСМА-80 — ЕСМА-85 ИЛПС, С5, I²C, D²B
Низкоскоростная (с селекцией)	Шина	ПЛ	1	РВ, ПП, ОП, ОП-КС	
Низкоскоростная (с узловой коммутацией)	Звезда	ПЛ	0,1	РВ с маршрутизацией	HDLC, X.25, DDCMP

Примечание: ВОЛС — волоконно-оптические линии связи; КК — коаксиальный кабель; ПЛ — проводная линия; РВ — разделение времени; ПП — передача приоритета; ОП — обнаружение передачи; ОП-КС — обнаружение передачи с контролем столкновений.

так как сообщение от источника передается в обоих направлениях к концам шины в широковещательном режиме. Узел приема осуществляет селекцию сообщений и запись данных в буферную память. Сети с такой топологией легко расширяемы. В кольцевой сети сообщения проходят от одного узла к другому по однонаправленным каналам передачи. Здесь также не требуется выбирать маршрут: каждый узел связи, подключенный к каналу передачи, передает сообщение по кольцу к узлу назначения. В таких сетях возникает проблема передачи управления от узла к узлу, необходимость решать, какой узел имеет в данный момент право на передачу сообщения. Эту функцию выполняет механизм управления передачей данных. Кольцевые СПД обладают потенциальной возможностью дальнейшего наращивания скорости передачи и децентрализации управления доступом к среде и передачей данных.

Методы управления доступом к среде. Разделение времени или сегментирование. Каждому узлу сети последовательно предоставляется право в течение выделенного времени передавать сообщения.

Передача приоритета или маркера. В СПД маркер последовательно переходит от узла к узлу и передает право на трансляцию сообщения тому из них, который имеет запрос от пользовательской системы. После передачи сообщения или пакета, маркер освобождается и переводится к следующему узлу сети.

Обнаружение передачи или случайный доступ. Все узлы сети ЭВМ постоянно включены на

Таблица 4

Характеристики типовых систем передачи данных

Фирма	Наименование сети	Максимальная длина канала, км	Физическая среда	Метод управления	Скорость передачи, Мбит/с	Максимальное число абонентов систем
Сверхвысокоскоростные СПД интегральных сетей						
AEG-Telefunken, Philips, Grundig und ent.	BIGFON	130	ВОЛС	РВ	560	Не регламентируется
Высокоскоростные СПД высокопроизводительных распределенных ВС						
Denelcor	HEP	0,1	КК, ВОЛС	РВ, ПП	100	64
Constar Communication	Hubnet	2	ВОЛС	РВ	50	65000
Control Data	LCN	0,9	КК	ПП	50	28
Network System Corp.	HYPERchannel	1	КК	ОП	50	256
NEC	C&C-Net	252	ВОЛС	ПП	32	126
Среднескоростные СПД локальных вычислительных сетей ЭВМ						
Proteon Associabes	proNET	2,5	КК,	ПП	10	255
Gandalf	PACNET	2,5	ВОЛС	ОП-КС	3,5	255
Godenoll Technology	Fiber Optic Net/One	2,5	КК, ВОЛС	ОП-КС	10	32
Optronics	Fibernet	2,5	ВОЛС	ПП	10	256
Amdax	Cablenet	140	КК	РВ	7/14	16000
Wang	Wangnet	3,7	КК	ОП-КС	9,6—12	512
Ungermann—Bass	Net/One	1,2	КК	ОП-КС	4/10	1024
Xerox	Ethernet	2,5	КК	ОП-КС	10	1024
IBM	Series/I—Ring	20	КК	ОП-КС	2	16
Низкоскоростные СПД малых локальных сетей						
Corvus Systems	Omninet	1,2	ПЛ	ОП	1	64 (микро-ЭВМ)
Zilog	Z-net	2,2	КК	ОП-КС	0,8	256
Zeda Computers Intern.	Infinet	5,5	ПЛ	ОП-КС	0,025	30 (микро-ЭВМ)
Philips	D²B	0,15	ПЛ	ПП	0,1	
Philips	I²C	0,01	ПЛ	ПП	0,1	
Низкоскоростные СПД локальных и региональных сетей ЭВМ с узловой коммутацией						
IBM	SNA	Не менее 10	ПЛ	Иерархический, протокол SDLC	0,056	Не регламентируется
Burroughs	BNA	Не менее 10	ПЛ	Децентрализованный, протокол BDLC, X.25	0,056	Не регламентируется
Computer Automation	Sy Fa	Не менее 10 и выше	ПЛ	Централизованный, протокол X.25	0,056	Не регламентируется
DEC	DECnet	Не менее 10 4,5	ПЛ КК	Децентрализованный, протокол DDCMP	0,019 1	250
Hewlett—Packard	DSN	Не менее 0,3	ПЛ	Децентрализованный протокол Bisync	0,019 2,5	Не регламентируется

прием данных. По адресу, передаваемому в начале кадра сообщения, можно определить, какому из устройств предназначено данное сообщение. Искажения, возникающие в канале при одновременной передаче сообщений несколькими узлами, обнаруживаются в узле приема по контрольным суммам.

Обнаружение передачи с контролем столкновений. Каждый передающий узел принимает свое сообщение и проверяет его с целью обнаружения столкновений с сообщениями от других узлов. Передача начинается при условии незанятости канала, поэтому столкновения могут происходить только в начале передачи, т. е. в течение того времени, которое необходимо на прохождение сигналов между двумя, одновременно начавшими передачу, станциями. При обнаружении столкновений обе станции на время прекращают передачу и повторяют ее через различные интервалы времени. Благодаря обнаружению столкновений в начале передачи и освобождению канала, повышается эффективность его использования.

Первые два метода чаще используются в кольцевых ЛВС. Для них характерна централизация управления доступом, благодаря чему исключаются столкновения сообщений в канале. Применение в СПД третьего и четвертого метода доступа позволяют строить ЛВС с шинной топологией [7, 8].

Протоколы передачи данных. Протоколы регламентируют программные процедуры управления доступом к среде и передачей сообщений в канал. Обеспечивают преобразование массивов данных, подготовленных для передачи в соответствии с принятой структурой транспортируемых кадров; взаимодействие станций при диалоге; последовательность передачи кадров сообщений и достоверность доставки информации [3, 7—9].

Устройство доступа к среде. ЭВМ к СПД подключается через коммутационный процессор транспортной станции. Соединение с физической средой осуществляется через устройство доступа к среде, которое выполняет функции канального и физического уровней архитектуры и реализует принятый метод доступа [7, 8, 10].

Характеристики типовых систем передачи данных приведены в табл. 4.

Распределенные вычислительные системы на основе ЛВС

Техника ЛВС позволила организовать распределенные ВС, обладающие следующими возможностями [1, 4, 7, 8, 10]:

создание единого центрального или нескольких основных банков данных;

обеспечение оперативного обмена информацией с центральным банком данных, со всеми ЭВМ сети, обрабатывающими устройствами и подсистемами;

использование вычислительных ресурсов всех подключенных к сети ЭВМ для распределенного решения отдельных задач;

дублирование вышедших из строя вычислительных средств;

одновременная параллельная обработка одной задачи на нескольких ЭВМ и сопоставление результатов об-

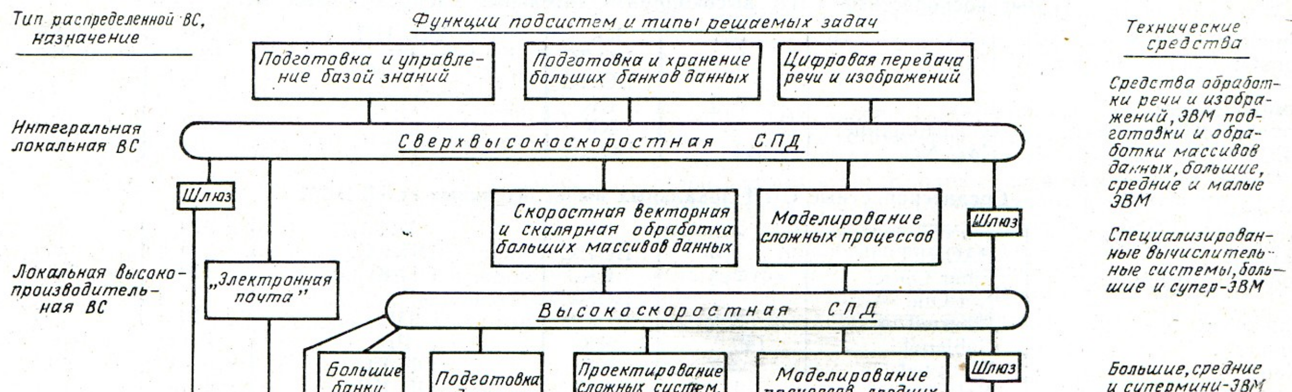


Рис. 3. Модель открытой интегральной сети

работки для обеспечения высокой надежности решения задачи;

доступность технических средств, которая снимает необходимость иметь все ресурсы на каждом вычислительном устройстве или в каждой подсистеме;

реконфигурация и программная настройка структуры ВС на тип и класс решаемой задачи;

автономизация подсистем или отключение их от сети при выходе из строя.

В локальных и региональных распределенных ВС, использующих *низкоскоростные СПД с узловой коммутацией,* организуются информационно-справочные системы с цифровой передачей больших массивов данных [6, 7]. В системах на основе *среднескоростных СПД* создаются распределенные информационно-справочные, вычислительные и управляющие системы, системы «электронной почты» и «безбумажного делопроизводства», системы распределенной подготовки и обработки данных и автоматизированного проектирования изделий малой и средней степени сложности [7, 8, 10]. Высокопроизводительные распределенные ВС, строящиеся с использованием *высокоскоростных СПД,* ориентируются на решение задач, связанных с обработкой больших массивов данных в реальном масштабе времени и применяются в САПР сложных систем, системах моделирования процессов газо- и гидродинамики, метеорологии [1, 5, 9, 12, 14].

Широкое распространение получают распределенные системы сбора данных и управления производством, контрольно-измерительных систем и систем автоматизированного управления технологическими линиями и процессами, гибких перепрограммируемых производств и роботизированных участков, цехов и заводов на основе *низкоскоростных СПД с упрощенными протоколами* взаимодействия, называемых м а л ы м и л о к а л ь н ы м и с е т я м и, для которых характерна передача информации небольшими массивами на расстоянии в несколько десятков и сотен метров [15].

В последнее время получили известность, так называемые, интегральные распределенные ВС, использующие *сверхвысокоскоростные СПД,* например система BIGFON (см. табл. 4). Эти системы объединяют большое количество разнообразных по классу и назначению ЭВМ и подсистем и обеспечивают решение разнообразных задач, характерных для проблемно-ориентированных систем и ЛВС.

Использование в распределенных ВС различных типов СПД предоставляет широкие возможности для достижения полной автоматизации производства и управления на всех уровнях — от постановки задачи, формирования инженерного решения, проектирования, изготовления промежуточной системы оснастки, до выпуска функционально законченного продукта производства. На рис. 3 представлена модель структуры открытой интегральной распределительной ВС, с использованием различных типов СПД. Общая структурная организация ВС строится по принципу подсетей, каждая из которых может создаваться на основе применения собственной топологии, аппаратных средств и протоколов обмена. Взаимодействие между подсетями осуществляется через шлюзовые ЭВМ (шлюзы), обеспечивающие аппаратное и программное сопряжение через интерфейсы и протоколы обмена, принятые в подсетях.

Последовательное развитие и широкое внедрение вычислительных сетей и СПД, обеспечивающих системную среду на всех уровнях открытой распределенной ВС, создает основу для применения распределенной обработки в АСУ и САПР, дает возможность приблизить средства обработки к объектам управления и успешно решить проблемы полной автоматизации на всех уровнях управления, проектирования и производства, расширяет области применения МСВТ и МПС, обеспечивает совершенствование методов управления народным хозяйством.

ЛИТЕРАТУРА

1. П о н о м а р е в Б. М, Д о м а р а ц к и й А. Н., Т о р г а ш е в В. А., Ш к и р т и л ь В. И. Локальные сети для автоматизации научных исследований, проектирования, производства и управления. Препринт 877. — Л.: АН СССР, ЛНИВЦ, 1982. — 21 с.

2. Б а к л а н о в А. В., Я к о в л е в С. А. Проектирование иерархической системы управления информационной сетью на базе машинных моделей. — В кн.: Синтез и проектирование многоуровневых систем управления. Ч. 2, Барнаул, 1982.

3. Д э в и с Д., Б а р б е р Д., П р а й с У., С о л о м о н и д е с С. Вычислительные сети и сетевые протоколы. — М.: Мир, 1982. — 562 с.

4. В е й ц м а н К. Распределенные системы мини- и микро-ЭВМ. — М.: Финансы и статистика, 1983. — 382 с.

5. А н д р е е в И. В., Б е р е з к и н Б. С., Б о г а т ы р е в В. И., К о т о в В. А. Распределенные системы обработки данных (РСОД). Препринт 5703.— М.: АН СССР, 1982. — 32 с.

6. Я к у б а й т и с Э. А. Архитектура вычислительных сетей. — М.: Статистика, 1980. — 279 с.

7. Я к у б а й т и с Э. А. Архитектура вычислительные сети. — М.: Финансы и статистика, 1984. — 232 с.

8. П р а н г и ш в и л и И. В., П о д л а з о в В. С., С т е ц ю р а Г. Г. Локальные микропроцессорные вычислительные сети. — М.: Наука, 1984. — 76 с.

9. С и п с е р Р. Р. Архитектура связей в распределенных системах. — М.: Мир, 1981 г., 746 с.

10. Х и н д и н Х а р в и Д ж. Многочисленные новые возможности локальных вычислительных сетей в 1980-х годах. — Электроника, 1982, № 2, с. 21—30.

11. Electronic Disign, 1984, v. 32, No 4, p. 180—194.

12. Ш а р е й к о Л. А. Проблема эффективности вычислительных сетей и пути ее решение. Препринт 3569. — М.: АН СССР, 1981. — 72 с.

13. Т у р б е р К е н н е т Д ж., Ф р и м е н Х а р в и А. Много изготовителей — много локальных сетей. — Электроника, 1982, № 2, с. 20—21.

14. К у п р и ч е н к о в а И. А., В и н о г р а д о в Б. Н. Организация программируемых ВС и ВС РОД в локальных сетях с высокоскоростными каналами передачи данных. Тез. докл. III Всесоюзной конференции «Вычислительные сети коммутации пакетов», с. 2, Рига: ИЭВТ АН Латв. ССР, 1983, с. 98, 99.

15. Г а л ь п е р и н М. П., М а с л е н и к о в Ю. А., Н е с т е р е н к о С. А., Р е з н и к А. Э. Архитектура малой локальной сети. — Электронная промышленность, 1983, Вып. 9, с. 66—68.

Статья поступила после доработки 2 июля 1984 г.

Уважаемые читатели!

Редакция журнала приносит свои извинения заинтересованным специалистам, чьи заявки на годовую подписку 1984 г. мы не смогли удовлетворить. Число наших подписчиков значительно превысило установленный на первый год издания тираж 7000 экз.

Одновременно сообщаем, что редакция журнала «МП» обратилась в Главное управление по распространению печати Министерства связи СССР с просьбой разрешить в дальнейшем подписку на журнал с любого очередного номера, т. е. поквартально.

УДК 681.327.8:537.533.3

М. А. Ананян, О. В. Мельникова

ОПТОВОЛОКОННЫЕ ЛОКАЛЬНЫЕ СЕТИ

В настоящее время длительность разработки и освоения систем информационного обмена в производстве становится соизмеримой с продолжительностью их жизненных циклов. В этих условиях повышение эффективности создания и применения систем определяется не только уровнем автоматизации процессов проектирования, но и степенью использования таких основных принципов построения, как распределенное управление системой, распределенные база данных и библиотеки программ и распределенная обработка данных.

Системы, основанные на этих принципах, получили общее название **информационно-вычислительных сетей.** Если их линейная протяженность невелика, т. е. все элементы сети расположены в пределах одного или нескольких близлежащих зданий, то такие сети обычно считают **локальными,** понимая под этим не только средства информационного обмена, но также совокупность вычислительных средств и оконечных устройств, используемых для ввода-вывода информации. В понятие сети не входят, однако, абоненты сети — источники и (или) получатели информации.

Локальные сети имеют ряд существенных особенностей, которые отличают их от более крупномасштабных информационно-вычислительных сетей:

— предназначены для решения задач в реальном масштабе времени;

— требуют линий связи и коммутационной аппаратуры невысокой стоимости, что приводит к необходимости применения микроЭВМ в узлах управления и использованию гораздо более простых и дешевых в реализации протоколов всех уровней;

— обеспечивают снижение задержки сообщений, благодаря небольшим размерностям сетей как по числу активных узлов, так и по линейным размерам, что увеличивает надежность правильной передачи, упрощает алгоритмы поиска ошибок и т. п. — все это создает предпосылки для разработки специализированных транспортных, сетевых и канальных протоколов, отличающихся от общеизвестных протоколов вычислительных сетей.

При проектировании локальных сетей может быть эффективно использован **метод отображения алгоритмов функционирования** на базовую структуру на основе типовых модулей. Этот метод приводит к поляризации процесса проектирования: основное внимание уделяется системному и технологическому уровням разработок при значительно меньших затратах сил и времени на промежуточных уровнях. Применение метода отображения алгоритмов приводит к созданию базовых, наиболее общих и устойчивых моделей сетей, которые варьируются в зависимости от особенностей данной задачи путем включения тех или иных специализированных модулей.

При переходе от системного уровня к технологическому представляется наиболее перспективным применением двух основных принципов физической реализации локальных сетей: ***цифрового представления сигналов*** всех видов, поступающих от различных источников информации и, в соответствии с этим, осуществления всех операций по переработке и передаче информации над цифровыми потоками; ***использования в качестве передающей среды оптоволоконных линейных трактов.***

Локальные сети, физическая реализация которых основана на этих двух принципах, наилучшим образом удовлетворяют основным требованиям, предъявляемым к сетям информационного обмена, и обеспечивают:

оптимальное использование оборудования физического, канального и сетевого уровней архитектуры;

гибкость по отношению к многообразию терминалов и протоколов обмена (прозрачность сети);

режимы скрытности передачи информации;

контроль и исправление ошибок;

высокую помехозащищенность и надежность;

возможность расширения функций и модернизации оборудования сетей без прокладки дополнительных каналов связи.

Можно выделить *три основных класса задач*, решение которых требует создания локальных сетей:

I. Распределенный сбор информации от большого числа разнообразных датчиков, обработка данных комплексом универсальных и специализированных ЭВМ, распределение информации по устройствам управления, отображение состояния объекта в целом и отдельных его подсистем в автоматизированных системах управления объектами, оборудованием, технологическими процессами.

II. Автоматизация конторской и управленческой деятельности при помощи локальных учрежденческих или так называемых интегральных сетей связи. Абоненты сети через «интеллектуальные» терминалы могут пользоваться общим банком данных, обмениваться между собой телефонной, видеотелефонной, символьной, графической информацией, организовывать конференции по телефонному и телевизионному каналам и т. д.

III. Автоматизированное проектирование разработок в многомашинных, в дальнейшем многопроцессорных, распределенных вычислительных сетях.

Типовая структура систем автоматизированного сбора, контроля и измерений первичной информации реализуется при использовании оптоволоконных систем сбора и передачи цифровых и аналоговых данных «Электроника МС 4101», «Электроника МС 8201» (см. цветной разворот).

Увеличение размерности сетей достигается применением **специализированных модулей — терминального** и **линейного концентраторов,** обеспечивающих комплексирование оптоволоконных систем. Терминальный концентратор позволяет снимать информацию со 128 датчиков, сгруппированных по территориальному признаку (рис. 1, *а*). Линейный концентратор осуществляет концентрацию территориально разнесенных (по группам) датчиков и дистанционный ввод информации в центральное устройство контроля и обработки (рис. 1, *б*).

Задачи автоматизированного управления объектами и технологическим оборудованием могут быть решены посредством локальной сети на основе оптоволоконных систем передачи и распределения цифровой и аналоговой информации «Электроника МС 4101» и «Электроника МС 8401». Распределение функций управления на значительной территории или по большому числу исполнительных устройств может осуществляться с помощью **распределителя оптоволоконных систем** (рис. 2). Перечисленные выше фрагменты: мультиплексированные оптоволоконные системы сбора, передачи и распределения цифровых и аналоговых данных, концентраторы и распределители информации являются базовыми структурными еди-

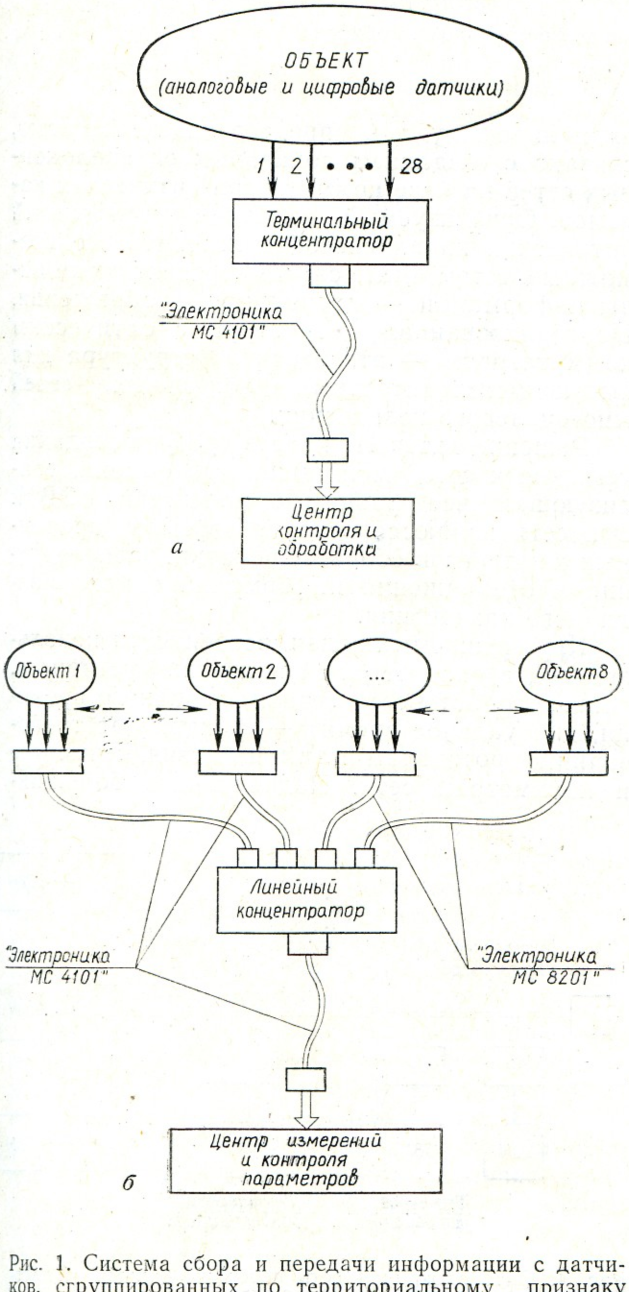


Рис. 1. Система сбора и передачи информации с датчиков, сгруппированных по территориальному признаку (*а*) с территориально разнесенных датчиков (*б*)

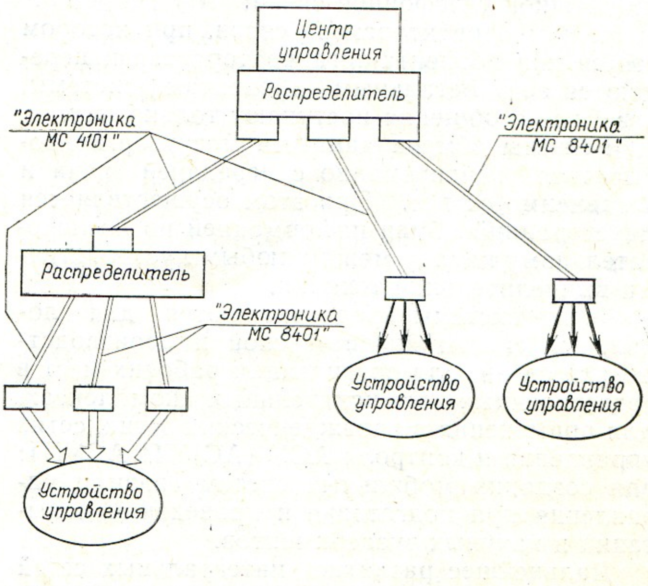


Рис. 2. Система распределения и передачи данных

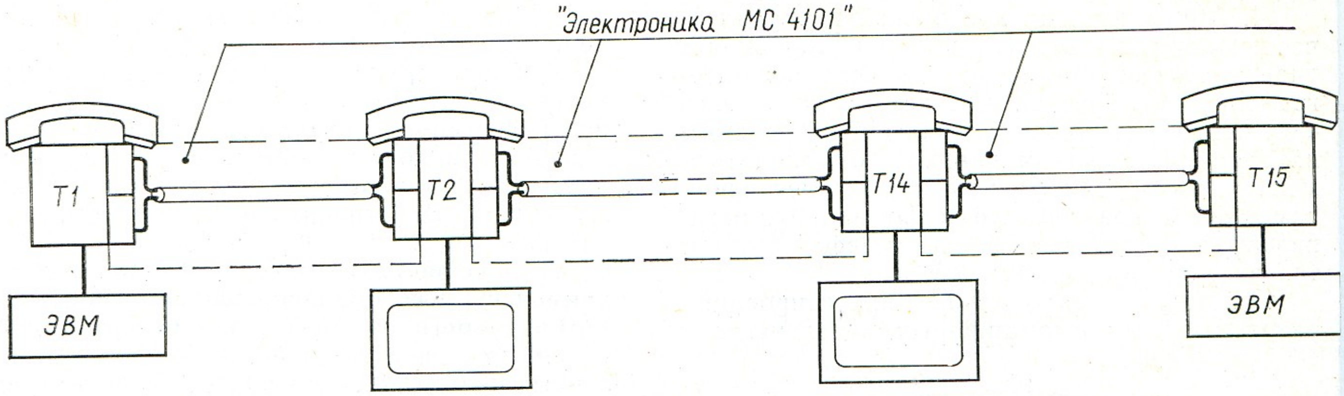


Рис. 3. Многотерминальная адресная оптоволоконная система информационного обмена

ницами, из которых пользователь может сформировать сколь угодно разветвленные локальные сети автоматизированных комплексов различного назначения.

Локальные сети, реализующие задачи II класса, обычно имеют структуру типа «звезда» с центральным коммутатором. Однако небольшие по числу абонентов (до 20) **интегральные системы,** предназначенные для одновременного и независимого обмена речевыми сообщениями и цифровыми данными, могут быть реализованы и на базе изделий «Электроника МС 4101» (рис. 3). Такие системы, разработанные в рамках общей программы развития оптоэлектронных локальных сетей, позволяют пользователю организовать два основных режима работы.

Режим обмена речевой информацией:

режим «каждый с каждым», адресный, система организации взаимодействия аналогична обычной телефонной связи;

режим «директорской» связи, при котором сообщения абонента-организатора связи передаются всем остальным абонентам системы, ответные сообщения поступают только ему.

Режимы обмена данными могут организовываться одновременно с передачей речи и независимо от нее. При этом осуществляется двусторонний обмен информацией по последовательному каналу между любыми устройствами вычислительной техники.

Такие системы предназначаются для локальных систем диспетчерской и производственной связи, для организации рабочих мест в разветвленных вычислительных комплексах, для применения в учрежденческих сетях, сетях управления и контроля АСУ (АСУТП, АСУП); для создания мобильных систем связи и управления при подготовке и проведении испытаний и научных экспериментов.

Дальнейшее развитие интегральных сетей связи как по числу абонентов, так и по качеству и видам услуг, предоставляемых сетью, **связано с созданием локальных оптоволоконных сетей на базе коммутатора оптических каналов.** Организуемый при этом непрерывный оптический канал способен передать все возможные, в том числе самые перспективные виды информации — подвижные изображения. Централизованная структура с оптическим коммутатором — **это базовая структура для высокопроизводительных локальных сетей многоцелевого назначения.**

Решение задач III класса требует создания локальных сетей межмашинного обмена, реализующих все уровни взаимодействия ЭВМ. Эти сети являются основой автоматизированных комплексов сбора, обработки, распределения и отображения информации самого различного назначения.

Круг решаемых задач, особенности используемых, в системе ЭВМ и других устройств, режимы работы сети (режим реального времени, виртуальное время) — формируют требования к протоколам взаимодействия, структуре и параметрам сети. Подавляющее большин-

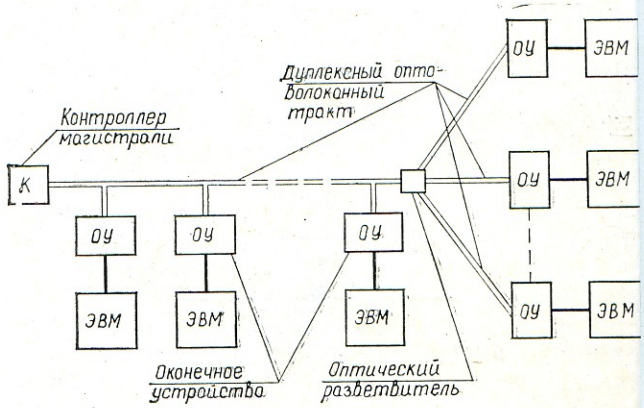


Рис. 4. Оптоволоконная сеть передачи данных «Электроника МС 8050»

ство задач, возникающих при создании АСУ, построении интегральных сетей на число абонентов 50...60, сетей межмашинного обмена для микроЭВМ, могут быть решены на основе разработанной оптоволоконной сети передачи информации «Электроника МС 8050» (рис. 4). Эта сеть предназначается для организации обмена массивами информации между территориально разнесенными устройствами посредством использования общей мультиплексированной оптоволоконной магистрали. Абонентами такой сети могут быть ЭВМ и их периферийные устройства, аппаратура сбора и распределения данных, терминалы учрежденческих или производственных локальных сетей.

В сети реализован централизованный сетевой протокол с управлением от контроллера магистрали. Число подключаемых к сети устройств — до 30, информация передается блоками по 32 слова (или менее), частота обращения к терминалу при этом составляет 50 раз в секунду.

Предполагается, что **локальные сети терминальных комплексов ЭВМ** типа «Электроника 60, 100/25, 79» и СМ-3, СМ-4, СМ-1300 будут организовываться на базе разрабатываемой в настоящее время унифицированной оптоволоконной системы информационного обмена, обеспечивающей подключение через оптоволоконный тракт к магистралям ЭВМ удаленных (до 500 м) внешних устройств, имеющих стандартные входы-выходы на интерфейсы ИРПР и ИРПС.

Таким образом, в настоящее время выпускаются типовые оптоволоконные структурные звенья, способные к взаимному сопряжению, на базе которых можно строить локальные информационные сети для терминальных комплексов ЭВМ, автоматизированных систем сбора, распределения и обработки данных, интегральные учрежденческие сети малой и средней емкости. Задача создания высокопроизводительных информационно-вычислительных сетей ЭВМ, включая сети для САПР, требует разработки протоколов канального и сетевого уровней, ориентированных на максимальное использование потенциальных возможностей оптоволоконной техники.

Необходимо отметить, что в настоящее время решение большинства задач перечисленных классов возможно на основе **оптоэлектронных локальных сетей типа «моноканал».** При этом может быть достигнуто максимальное упрощение аппаратуры доступа к сети, получаемое за счет использования потенциальной широкополостности оптического кабеля. В целом, это увеличивает надежность сети при общем снижении затрат на ее разработку, внедрение и эксплуатацию. Однако, с учетом специфических особенностей локальных сетей, применение при их проектировании известных сетевых и транспортных протоколов, разработанных для крупномасштабных сетей, представляется неоптимальным. Сетевые протоколы для оптоэлектронных локальных сетей должны быть разработаны с учетом следующих требований:

реализация блока доступа к сети в основном аппаратными средствами (этого требуют высокие, до 140 Мбит/с, скорости передачи информации по оптоволоконному тракту);

возможность побитовой обработки управляющей информации в целях сокращения объема аппаратуры блоков доступа к сети;

при разработке протоколов сетевого и канального уровней целесообразно взять за основу отработанные принципы взаимодействия узлов и блоков ЭВМ.

В заключение следует подчеркнуть наиболее существенные особенности оптоволоконных локальных сетей:

1. Оптоволоконные сети «прозрачны» по отношению к любым сетевым и канальным протоколам, если эти протоколы реализованы в узлах сети.

2. Свойства передающей среды — оптического кабеля — оказывают значительное влияние на выбор протоколов более высоких уровней иерархии, структуру и архитектуру локальной сети.

3. Проектирование и разработка оптоволоконных локальных сетей требуют выбора наиболее рациональных структур и архитектур (систем протоколов) с учетом специфических особенностей оптоволоконных линейных трактов.

Статья поступила 15 октября 1984 г.

РЖ ВИНИТИ

Роль систем обработки текстов в автоматизации учреждений и перспективы их применения в Индии. Role of wordprocessors in office automation and their prospects in India. Tripathi R. C. «Electron.-If. and Plann.», 1982, 9, № 10, 530-549 (англ.).

К новым видам учрежденческой автоматики относятся факсимальные устройства и системы обработки текстов (СОТ), причем последние оказались наиболее эффективным средством повышения производительности труда конторских работников. Рассматриваются перспективы использования СОТ в Индии и других странах. Проводится анализ мирового рынка в области СОТ.

Предполагается, что в течение 1980—1985 гг. из всех видов оборудования ВТ наибольшим ростом производства и сбыта будут отличаться СОТ. Приводятся спецификации автономной СОТ, выпускаемой с начала 80-х годов в Индии на базе МП фирмы Intel.

УДК 681.327.8

В. М. Долкарт, С. Л. Карпенко, Г. М. Курцман, Е. Е Перцов, С. Ф. Редина

ДИСТАНЦИОННАЯ МАГИСТРАЛЬ — ЛОКАЛЬНАЯ УПРАВЛЯЮЩАЯ ВЫЧИСЛИТЕЛЬНАЯ СЕТЬ МИКРОСИСТЕМ В7/В9

Дистанционная магистраль обеспечивает обмен данными между микросистемами МС УВТ В7/В9, объединенными в локальную управляющую вычислительную сеть.

Локальная сеть микросредств управляющей вычислительной техники МС УВТ В7/В9 [1, 2] предназначена для создания управляющих распределенных систем производства или одного объекта, рассредоточенного на расстоянии 0,5—1,5 км.

Станции локальной управляющей вычислительной сети (ЛУВС) — управляющие микросистемы В7/В9: их конфигурация варьируется в соответствии с назначением.

Ряд управляющих микросистем В7/В9 объединяется в единую локальную управляющую вычислительную сеть с помощью общей последовательной двунаправленной **дистанционной магистрали** (ДМ) с моноканальной связью на большие расстояния (для необходимого обмена информацией между всеми станциями системы). ДМ В7/В9 должна:

передавать данные с высокой скоростью, необходимой для управления в реальном масштабе времени;

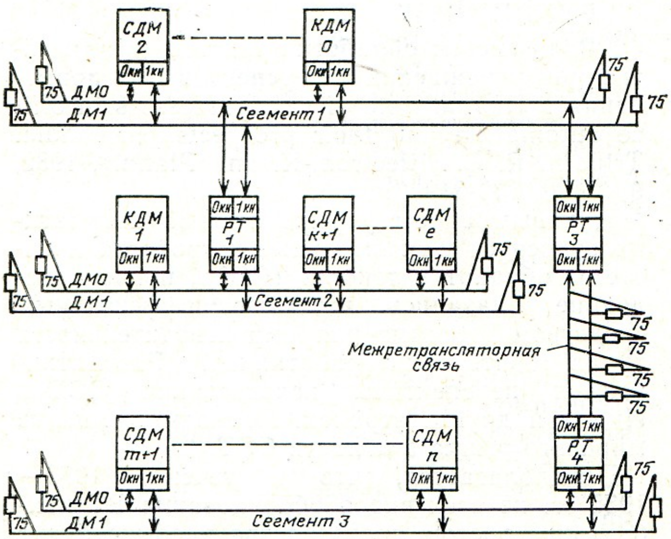


Рис. 1. Подключение микросистем к дистанционной магистрали локальной управляющей вычислительной сети В7/В9

передавать данные с повышенной надежностью;

позволять гибко модифицировать систему за счет подключения (отключения) новых станций и объединения сетей (без останова всей сети);

распараллеливать и совмещать во времени процессы сбора, обработки данных в микросистемах и организации доступа к ДМ.

Интерфейс микросистем локальной сети В7/В9 с дистанционной магистралью (рис. 1) — станции ДМ (СДМ) — входят в микросистемы. В состав СДМ входят три модуля (МС УВТ В7/В9 в стандартных конструктивах): одноплатная микроЭВМ на базе микропроцессора КР580ИК80А; устройство управления интерфейсом ДМ, подключаемое к микроЭВМ через внутримашинную магистраль, и устройство сопряжения с приемопередатчиками ДМ, работающими непосредственно на линию передачи магистрали.

Управление работой ДМ — централизованное: одна из станций работает в режиме контроллера ДМ (КДМ). Единое центральное устройство управления ДМ облегчает опрос требований на приоритет.

Для повышения надежности ЛУВС В7/В9 используется дублированная дистанционная магистраль.

Для ДМ выбран эстафетный принцип передачи прав доступа к моноканалу с использованием метода коммутации пакетов сообщений разделением их во времени.

Доступ станций к моноканалу разрешается после получения специальной разрешающей команды «передачи эстафеты», называемой ВЫЗОВ. Передает информацию только одна станция в одно время. Это уменьшает искажение передаваемых сигналов и соответственно повышает надежность работы ЛУВС.

Контроллер ДМ, периодически опрашивая по ДМ статус станции (или с помощью системы прерываний), определяет очередность предоставления магистрали запрашивающим станциям и эстафету для текущего периода. На каждом ее этапе проводится сеанс связи между двумя СДМ, в котором первичная станция руководит обменом данных со вторичной станцией. При этом ведется широко-

вещательная, без выбора маршрута передача пакетов информации во все станции в изохронном (последовательном, побайтно синхронизированном) режиме.

При проектировании **протокола ДМ** использована общепринятая концепция многоуровневой структуры протоколов. Реализованы 1, 2 и 4 нижние уровни эталонной модели открытых систем международной организации по стандартизации [3].

Первый уровень протокола управления передачей данных по дистанционной магистрали (УПДМ) определяет физические связи и электрические характеристики ДМ. В рамках проектирования первого уровня протокола выбраны линия передачи ДМ, приемопередатчики ДМ для приема-передачи сигналов по линиям передачи, определены максимальная скорость передачи сигналов, предельная нагрузка и длина линий передачи, правила подключения к магистрали, при которых обеспечивается высокая помехозащищенность передаваемых сигналов.

Линия передачи (передающая среда) ДМ — 75-омный коаксиальный кабель. Линия передачи последовательно связывает все станции сети и имеет электрическое согласование на обоих концах. Приемопередатчики сигналов ДМ подключаются к ней пассивными отводами в виде коротких отрезков коаксиального кабеля. Линия передачи ДМ может содержать несколько сегментов (см. рис. 1). Между сегментами обмениваются сигналами ретрансляторы сигналов (РТ), выполняемые на базе приемопередатчиков ДМ.

Приемопередатчик ДМ (рис. 2) на базе микросхем серий К155, К293, совместим по входам, подключаемым к микросистемам, с логикой ТТЛ. Передатчик ДМ имеет защиту от скачков напряжения и от коротких замыканий в коаксиальных кабелях ДМ. Приемник ДМ содержит цепь автоматической регулировки порога срабатывания. Она настраивается в соответствии с уровнями сигналов в линии передачи в месте подключения станции.

Оптроны в трактах передачи и приема сигналов гальванически развязывают станции и

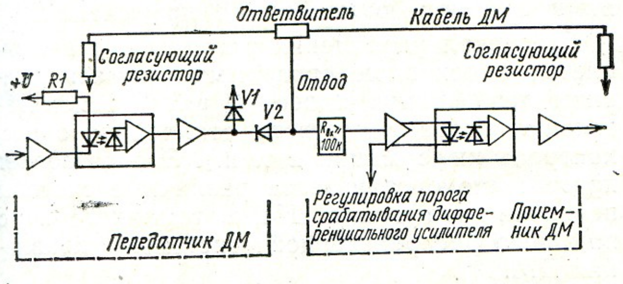


Рис. 2. Структурная схема приемопередатчика ДМ

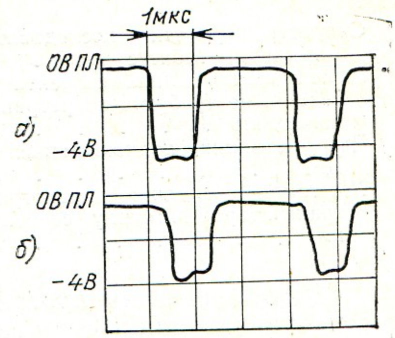


Рис. 3. Осциллограмма напряжения на входе и выходе линии передачи ДМ длиной l=500 м: *а* — вход линии, *б* — выход линии

ДМ. С этой же целью электропитание выхода передатчика и входа приемника, непосредственно связанных с линией, — плавающее.

Приемопередатчики ДМ при отключении и при приеме представляют собой для кабеля ДМ высокое входное сопротивление, очень мало нагружающее кабель. Это исключает искажение передаваемых по нему сигналов. Искажения из-за отражений устраняются согласованием кабеля на обоих концах (рис. 3).

Сигналы информации и управляющие команды передаются по ДМ в виде группы слогов (содержат байты данных, сформированные в пакет). Каждый слог пакета содержит 11 битов (рис. 4): стартовый бит «Лог. 0», 8 информационных битов байта данных, бит контроля по нечетности и стоповый бит «Лог. 1». Модуляция сигналов — уровневая, без возврата к нулю.

При приеме пакета синхронизация — изохронная, т. е. с собственной частотой, формируемой станцией. Синхронизация ведется по стартовому биту каждого слога, с контролем наличия стартового бита в его середине и с приемом всех остальных разрядов слога (синхронно с серединой стартового бита слога).

При отсутствии отражений искажения битовой посылки в коаксиальном кабеле зависят, в первую очередь, от длины кабеля.

В соответствии с характеристиками приемопередатчиков ДМ и допустимым уровнем искажений сигналов в линиях передачи согласно первому уровню протокола локальной сети В7/В9 реализованы следующие основные электрические характеристики ДМ:

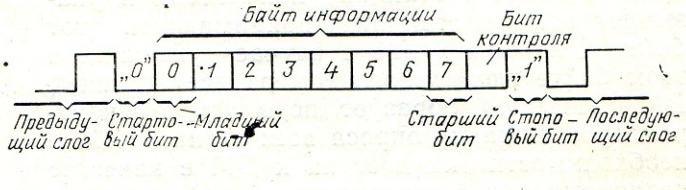


Рис. 4. Формат слога информации ДМ

Скорость передачи последовательной информации, Мбит/с	1
Максимальная длина сегмента линии передачи, не требующего ретрансляции сигналов, км	0,5
Число станций, подключаемых к сегменту ДМ	2—64
Максимальная длина магистрали (с ретрансляторами между сегментами ДМ), км	1,5

Второй уровень протокола УПДМ (уровень канала данных) задает функции управления передачей по ДМ: адресацию станций; определение форматов отдельных слогов и пакетов передаваемой информации, форматов специализированных служебных команд; контроль передаваемой информации.

Согласно второму уровню протокола выделены такие параметры: адресация — длина адреса 7 бит, число адресуемых станций 127, наличие общего глобального адреса для всех станций; длина пакета — однослоговый для служебных команд, управляющих организацией эстафеты; многослоговый (5—10 000 слогов) для данных; контроль — послоговый по нечетности, по стартовым и стоповым битам и пакетный по сигнатуре массива данных.

В общем случае полный пакет данных, передаваемый по ДМ, состоит из трех полей: A — поле адреса, $ИУ$ — информационно-управляющее поле, $К$ — поле контроля.

Длина поля A — 1 байт данных. Старший разряд байта адреса А определяет однослоговый пакет (при «0» значении бита) или многослоговый пакет (при «1» значении бита); 7 младших битов байта номера станции А указывают непосредственно сам номер станции или контроллера от 000 0000 до 111 1110. Номер 111 1111 — глобальный номер — относится ко всем станциям одновременно.

Поле $ИУ$ содержит адрес станции источника, данные и управляющую информацию транспортного уровня протокола УПДМ, определяющую функционирование распределенной локальной управляющей вычислительной сети.

Поле $К$ длиной в 2 слога служит для передачи 16-разрядной аппаратно формируемой контрольной сигнатуры массива передаваемых данных.

Однослоговый пакет содержит только байт поля A (адреса станции) получателя и указывает номер станции, принимающей этот байт. Контроллер ДМ в случае указания в этом байте глобального номера всех станций ДМ (111 1111) образует посылаемую контроллером команду опроса всех станций ДМ о необходимости их доступа к ДМ в качестве первичных или о готовности их работы в качестве вторичных.

Ответы станций ДМ на команду опроса контроллера ДМ выдаются через количество слогов, совпадающее с номером отвечающей станции ДМ. Каждая станция имеет программно настраиваемый счетчик задержки ответа станции на команду опроса.

При указании в байте однослогового пакета номера какой-либо станции образуется команда вызова станции ДМ. Она определяет станцию с данным номером в качестве первичной в последующем сеансе связи (передает эстафету данной станции).

Функции *четвертого (транспортного) уровня протокола* обеспечивают высокую надежность алгоритмов обмена данными через ДМ, т. е. надежность локальной сети. На этом уровне протокола:

надежность передачи данных повышена повторением передачи в случае ошибок; правильность передачи и приема данных на этом уровне контролируется с помощью квитанции [4, 5];

управление работой станций сети — супервизорное с разделением времени магистрали между станциями;

обеспечены целостность данных, принятых станцией и управление потоком датаграмм в ДМ;

станции ДМ для работы в режиме опроса настраиваются автоматически;

в исходное состояние устанавливаются все станции сети;

переключение канала магистрали, выбор для работы основного или резервного канала — программные.

В контроллере ДМ реализована служба реального времени. Во всех станциях сети установлен и поддерживается единый отсчет времени.

Протокол управления работой ДМ обеспечивает достаточно высокую скорость обмена данными между станциями сети. Типичные времена t сеанса связи в зависимости от длины N пакета данных — 9, 10 и 56 мс соответственно при $N = 100$, 200 и 400 байт.

В протоколе ДМ локальной сети В7/В9 реализован также интерфейс транспортного и более высоких уровней. Это обеспечивает асинхронный обмен сообщениями между уровнями с помощью механизма прерываний.

Протокол управления работой ДМ выполняется в основном программно, за исключением дешифрации адреса и ряда вспомогательных операций (ответы станций на опрос, контроль информации и т. п.). Это позволяет программно организовать различные способы передачи эстафеты в ДМ, в том числе и с помощью распределенной системы управления ДМ.

В настоящее время управляющая вычисли-

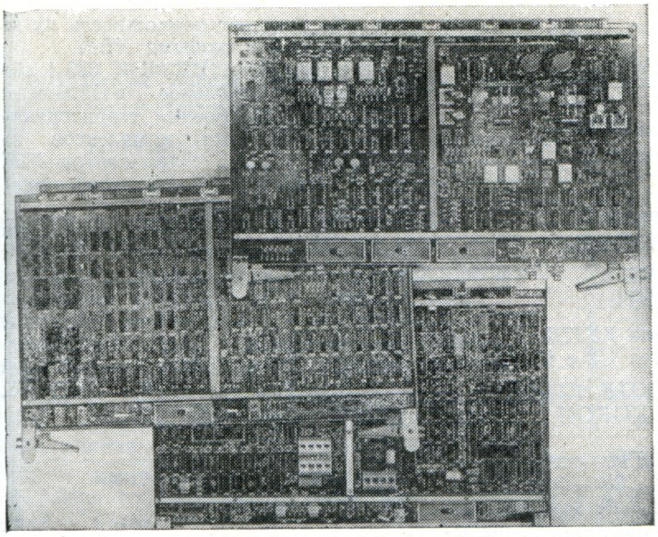


Рис. 5. Комплект модулей станции ДМ

тельная сеть В7/В9 оснащена необходимой аппаратурой (рис. 5) и программным обеспечением для практического использования в различных управляющих системах. Созданы специальные устройства сопряжения (с аппаратно-программным управлением) для станций и контроллера ДМ, система конструктивов для линий передачи ДМ, набор сервисных программ. Опыт применения локальной сети В7/В9 показал ее полную пригодность для построения систем с распределенной управляющей вычислительной мощностью.

ЛИТЕРАТУРА

1. Шереметьевский Н. Н., Долкарт В. М., Новик Г. Х., Лукьянов Л. М., Пурэ Р. Р. Микросредства управляющей вычислительной техники серии В7: Средства автоматизации управления электротехническими системами и изделиями. — М., 1980. (Тр. ВНИИЭМ, т. 62).

2. Шереметьевский Н. Н., Долкарт В. М. Магистрально-модульные микросредства управляющей вычислительной техники. — Микропроцессорные средства и системы, 1984, № 2, с. 24—27.

3. Якубайтис Э. А., Баумгарт В. Ф. и др. Архитектура локальных вычислительных сетей. — Автоматика и вычислительная техника, 1983, № 2, с. 3—20.

4. Дэвис Р., Барбер Б., Прайс У., Соломонидес С. Вычислительные сети и сетевые протоколы. М.: Мир, 1982.

5. Stieglitz. Local Network Access Tradeoffs. — Computer Design, 1981, N 10, pp. 163, 164, 166, 168.

Статья поступила 23 апреля 1984 г.

УДК 681.326.3:681.327.8

А. А. Мячев, А. А. Снегирев

КОНТРОЛЛЕРЫ ПРИБОРНЫХ ИНТЕРФЕЙСОВ мини- и микроЭВМ

Одной из наиболее важных проблем построения автоматизированных систем научных исследований (АСНИ) является выбор базовых технических и программных средств для автоматизации наиболее распространенных экспериментов с учетом рационального сопряжения с экспериментальными установками и приборами и организации оперативного (в реальном времени) интерактивного режима [1, 2].

При выборе стандартных приборных интерфейсов для сопряжения экспериментальных установок с мини- и микроЭВМ принимают во внимание такие показатели как гибкость, быстродействие, широкие функциональные возможности и распространенность, проработанность основных разделов и технических характеристик стандарта [2, 3].

В современных АСНИ подсистемы связи с объектом и экспериментальным оборудованием (ПСО) реализуются на основе машинно-независимых стандартных приборных интерфейсов, главным образом, САМАС (КАМАК) и МЭК 625-1. При создании вычислительных и управляющих средств АСНИ на базе ЭВМ возникает проблема организации эффективного взаимодействия центрального ядра АСНИ и ПСО. На схемном уровне проблема решается разработкой контроллеров приборных интерфейсов для ЭВМ, обеспечивающих сопряжение магистрали ввода-вывода управляющей ЭВМ с магистралями периферийных подсистем (ППС).

Промышленной основой АСНИ конкретного назначения на базе СМ ЭВМ являются измерительно-вычислительные комплексы (ИВК), входящие в класс проблемно-ориентированных комплексов (ПОК) [1, 4].

При проектировании ИВК, соответствующих требованиям современных АСНИ, необходимо создавать функционально полный набор контроллеров приборных интерфейсов на основе анализа их номенклатуры, функциональной и структурной организации и рационального выбора критериев эффективности сравниваемых вариантов [4, 5].

Анализ средств сопряжения магистралей

При анализе АСНИ представляется в виде системы, содержащей набор магистральных подсистем, объединенных между собой средствами сопряжения. Каждая подсистема характеризуется собственным магистральным интерфейсом, например, общая шина (ОШ), магистраль крейта КАМАК, канал общего пользования (КОП), к которым подключаются функциональные средства ввода, вывода, обработки и хранения информации. Особенность средств сопряжения состоит в организации взаимодействия между любыми устройствами сопрягаемых подсистем независимо от их конкретного назначения и реализации.

Вариантом концептуальной модели средств сопряжения является их представление в виде блоков связи (БС), подключенных к каждой из сопрягаемых подсистем, объединенных между собой каналом связи (КС).

Классификация средств сопряжения по основным показателям приведена в табл. 1.

К классу **гомогенных** (сопрягающих функционально идентичные интерфейсы) относятся различные *внутрисистемные средства,* расширяющие функциональные возможности вычислительного и управляющего ядра АСНИ.

Таблица 1

Классификация средств сопряжения магистралей

Наименование параметра	Группа средств сопряжения	Пример
Соотношение интерфейсов сопрягаемых подсистем	Гомогенные Гетерогенные	РИФ, СГИ, УСОШ ОШ-КАМАК, ОШ-МЭК 625-1
Структура объединения подсистем	Двухточечное Многоточечное	КК, ККПД, КПВ, КПИ
Конструктивная реализация	С внутренним КС С внешним КС	КК типа ККПД КК типа U2 с КПД
Принцип передачи информации в КС	С параллельным КС С последовательным КС	КК, ККПД, КПИ КПМ, КПКОП

Примечание: РИФ — расширитель интерфейса, СГИ — сегментатор интерфейса, УСОШ — устройство согласования общин шин, 2К — тип интерфейса СМ ЭВМ; КК — контроллер крейта, ККПД — КК с каналом прямого доступа; КПВ — контроллер параллельной ветви; КПМ — контроллер последовательной магистрали, КПИ — контроллер приборного интерфейса, КПКОП — контроллер последовательного КОП.

Таблица 2

Структурно-функциональные типы контроллеров

Принцип подключения	Тип КС	Назначение
Радиальный	Внутренний, параллельный	Локальные ИВК
Радиальный	Внешний, параллельный	Локально-распределенные ИВК с небольшим числом ППС
Магистральный	Внешний, параллельный	Локально-распределенные ИВК
Радиальный	Внешний, последовательный	Территориально - распределенные ИВК высокого быстродействия
Магистральный	Внешний, последовательный	Территориально - распределенные ИВК

К **гетерогенным** (сопрягающим функционально различные интерфейсы) относятся *контроллеры приборных интерфейсов и устройства согласования различных системных интерфейсов.*

Средства сопряжения, обеспечивающие двухточечное соединение, предназначены для объединения двух подсистем (радиальное подключение к ядру АСНИ). Средства, обеспечивающие многоточечное соединение, служат для объединения трех и более МПС (групповое, обычно магистральное, подключение к ядру АСНИ).

Средства сопряжения *с внутренним КС* реализуются в виде единого блока и КС составляют внутриплатные и межплатные соединения. Средства сопряжения *с внешним КС* выполняются в виде двух или более конструктивно законченных блоков, объединенных кабельным соединением.

В *параллельных КС* используется принцип пространственного кодирования, т. е. для каждого логического сигнала КС существует физический канал (линия связи). В последовательных КС применяется принцип временного кодирования, т. е. различные логические сигналы передаются по одним и тем же физическим каналам КС.

В результате можно выделить пять основных практически целесообразных структурно-функциональных типов контроллеров (табл. 2). Структурно полный набор контроллеров обеспечивает рациональное построение АСНИ произвольной конфигурации. Для группы контроллеров ОШ—КАМАК такой набор составляют: КК радиального подключения к ОШ, драйверы параллельной и последовательной ветви КАМАК. Драйверы целесообразно создавать в конструктивах приборных интерфейсов, что обеспечивает возможность их использования с любыми ЭВМ.

Контроллеры ОШ-МЭК 625-1 обеспечивают локальное подключение ППС. При построении территориально-распределенных комплексов используют либо ретранслятор МЭК 625-1, либо последовательный вариант его реализации.

С учетом выявленных требований построения разнообразных АСНИ с рациональной архитектурой и распределения по областям применения оптимальна следующая временная приоритетность разработки универсальных контроллеров минимальной номенклатуры для ЭВМ с ОШ: КК, ККПД, КПВ, КПМ.

Показатели назначения контроллеров

При проектировании ИВК первой очереди основное внимание уделяется оптимизации архитектуры контроллеров радиального подключения к ОШ (табл. 3). Выбранный режим определяет состав функциональных блоков и узлов контроллера. Для работы по ПК необходимы блоки дешифрации адресов и команд ОШ, адресуемых и буферных регистров, обработки запросов и прерывания, формирования команд приборного интерфейса, согласования с линиями магистралей (приемопередатчики). Число адресуемых регистров выбирается от двух до четырех в зависимости от типа приборного интерфейса и соответствующих схемно-реализуемых функций.

Для работы в режиме ПДП к этому набору добавляются блоки захвата ОШ и управления операциями ПДП. Число адресуемых регистров увеличивается до пяти-восьми. При реализации нескольких независимых подканалов ПДП в одном контроллере каждому подканалу отводится свой набор адресуемых регистров и в контроллер добавляется узел управления запуском подканалов.

При непосредственной передаче полной команды по линиям ОШ обеспечивается наиболее быстрый доступ к информации в регистрах ППС, однако этот способ требует значительной части адресного пространства, что снижает возможности подключения дополнительного периферийного оборудования и блоков ОЗУ. Он применим при малой разрядности команды приборного интерфейса и для относительно небольших и дешевых систем.

При передаче команды через адресуемые регистры контроллера обеспечивается минимальное использование адресов системы, однако это наиболее медленный способ формирования команды приборного интерфейса. На практике он используется для создания сложных системных конфигураций с большим набором дополни-

Таблица 3

Показатели, определяющие функциональные характеристики контроллеров

Показатель	Группа параметров	Зависимые функциональные характеристики
Режим информационного обмена	Программный Режим ПДП	Пропускная способность
Способ формирования команды	Непосредственный параллельный Косвенный параллельный Комбинированный	Время инициации обмена, пропускная способность по ПК
Способ арбитража и формирования векторов прерывания	Индивидуальные вектора Групповые вектора Единственный вектор	Реактивность, пропускная способность по ПК

тельного периферийного оборудования и схемно-реализуемых функций.

При комбинированном способе полная команда разделяется на поля и через адресуемые регистры контроллера передаются достаточно редко изменяемые части команды. Экономия адресов системы гарантирует более быстрый доступ к регистрам ППС и в значительной степени облегчает программирование ППС [5].

При формировании индивидуальных векторов прерывания для каждого из функциональных устройств ППС обеспечивается наибольшая реактивность при наибольших затратах по оборудованию контроллера и объему занимаемой области векторов прерывания в ОЗУ.

Способ формирования вектора для группы запросов — своего рода компромисс между затратами и реактивностью системы. Приоритетное обслуживание групп запросов — *схемное*, запросов функциональных устройств одной группы — *программное*. Обеспечение эффективности варианта связано с необходимостью решения задачи выбора достаточного числа групп запросов и оптимального разбиения запросов функциональных устройств конкретной системы по группам, в зависимости от требуемых параметров их обслуживания.

Динамическое изменение приоритетов различных запросов и групп обычно осуществляется маскированием отдельных прерываний.

Структурная организация контроллеров

Анализ структурной организации контроллеров базируется на представлении технических средств в виде композиции автоматов: операционного (ОА) и управляющего (УА). ОА выполняет функции хранения, преобразования и передачи информации между каналами связи сопрягаемых магистральных интерфейсов. Структуру ОА образует набор операционных элементов (регистров, счетчиков, шифраторов, дешифраторов, коммутаторов) и связей между ними. УА организует последовательность действий операционных элементов, реализуя общий алгоритм функционирования ОА. Связи между автоматами составляют управляющие сигналы, а также логические условия, используемые в УА при выполнении условных переходов.

Общая конфигурация ОА определяется принципом реализации запоминающих элементов внутренних регистров контроллера. Возможно использование:

1) индивидуальных регистровых схем по числу внутренних, в том числе адресуемых, регистров контроллера. В этом случае принципиально достижимо максимальное быстродействие работы ОА, так как обеспечивается параллельный доступ ко всем запоминающим элементам контроллера *(линейная регистровая структура)*. С учетом возможных ограничений этот принцип применим при относительно небольшом числе внутренних регистров контроллера и используется в том случае, когда информационный обмен с устройствами ППС обеспечивается только по ПК;

2) общего блока памяти адресуемых регистров (кольцевые регистровые структуры). Выполнение операций, как правило, требует извлечения содержимого отдельных адресуемых регистров из БП, его модификации и последующего сохранения в блоке памяти. Вариант эффективен при большом числе адресуемых регистров, в первую очередь, при реализации обмена в режиме ПДП по нескольким независимым подканалам. Длина регистрового кольца (число соединенных регистров) определяется объемом информации, хранимой в адресуемых регистрах и необходимой для выполнения наиболее сложной операции.

Используются два основных варианта построения УА: схемное («жесткая» логика); микропрограммное (МПУ).

С учетом разнообразия управляющих функций контроллеров и сложности их унификации *при схемном управлении* обычно используются микросхемы малой и средней степени интеграции. Это рационально для относительно простых контроллеров и максимального быстродействия работы узлов. При глубокой проработке и высокой степени стандартизации логической организации интерфейсов сопрягаемых ППС можно унифицировать управляющие функции и реализовать их на основе специализированных БИС. Схемное управление эффективно в контроллерах любой сложности, использующих интерфейсные БИС.

При микропрограммном управлении формирование последовательности управляющих сигналов осуществляется с помощью ПЗУ.

Структурно микропрограммный УА представляет совокупность узлов: генератор — счетчик — ПЗУ — регистр. Счетчик ведет последовательный перебор номеров тактов, а ПЗУ используется для дешифрации управляющих сигналов, соответствующих каждому такту при выполнении определенной операции и запоминаемых далее в регистре. Схема проста логически и целесообразна в контроллерах средней сложности, например, с одним подканалом ПДП [5]. В более сложных случаях ПЗУ используется и для выбора номера следующего такта (состояния УА) в зависимости от логических условий и номера текущего такта. Схема универсальна, пригодна для реализации большого числа разнообразных режимов обмена.

Таблица 4

Относительные схемные затраты вариантов реализации ОА и УА

Структура ОА	Исполнение УА	Затраты оборудования: Обмен по ПК	Затраты оборудования: Обмен по ПК и ПДП
Линейная	Схемное	1	2
Линейная	Микропрограммное	1,25	1,9
Кольцевая	Схемное	1,3	1,8
Кольцевая	Микропрограммное	1,35	1,4

Возможна индивидуальная настройка контроллеров на специфические режимы по требованиям конкретного применения. Однако схема является и наиболее сложной логически. Ее разработка сравнима с разработкой блока МПУ процессора микроЭВМ.

Анализ аспектов структурного построения ОА и УА контроллеров показывает, что при реализации в контроллере только ПК для доступа к функциональным устройствам ППС целесообразна линейная регистровая структура со схемным управлением. В случае реализации контроллером различных вариантов режима ПДП целесообразна кольцевая регистровая структура с МПУ (табл. 4).

Критерии и методы оценки эффективности контроллеров

Принятие обоснованных технических решений при разработке архитектуры контроллеров приборных интерфейсов связано с комплексной оценкой их качества и предполагает выбор и обоснование критериев эффективности их функциональной и структурной организации.

В большинстве случаев в качестве критериев эффективности принимаются некоторые основные из обобщенных технико-экономических показателей продукции (характеристики производительности, интерпретируемые для контроллеров показателями пропускной способности, реактивности, стоимости, надежности и т. д.).

Основные назначения контроллеров — организация информационного обмена с ПУ подключаемой ППС, эффективность которого наиболее полно отражается в показателях максимальной $\Pi_м$ и средней $\Pi_с$ пропускной способности.

Выбор в качестве критерия только $\Pi_м$ целесообразен при проектировании специализированных систем с заданным классом решаемых задач. В этом случае оценивается быстродействие схемных средств в основном режиме функционирования, что не позволяет объективно отразить архитектурные особенности контроллеров, которые по существу выполняют функции группового канала, организующего информационную связь между ОЗУ и абонентами ППС.

Более полно архитектуру контроллеров отражает показатель $\Pi_с$, усредняющий пропускную способность контроллера по возможному числу абонентов ППС. Недостатком критерия $\Pi_с$ является сложность вычислений и возможная неоднозначность построения модели оценки искомых характеристик.

В случае удовлетворения сравниваемых вариантов заданным функциональным ограничениям выбор далее может проводиться по критериям стоимости, надежности, технологичности и т. д. в зависимости от специфики проектируемой системы.

Таблица 5

Основные показатели оценки эффективности контроллеров

Показатель (фактор)	Ранг	Поддиапазоны	Оценка
		Производительность	
Режим обмена	10	Только ПК	1
		ПК+1 КПД	2
		ПК+(2...4) КПД	2,5
		ПК+(4...8) КПД	3
Число уровней прерывания	9	1...2	1
		3...4	1,5
		8...16	2
		24 и более	2,5
Принцип формирования команды ППС	8	С делением на поля	2
		Косвенный (регистровый)	1
		Прямой	3
		Программное обеспечение	
Операционные системы (ОС)	10	Реального времени	3
		Диалоговая ОС	2
		Дисковая ОС	1
Языки программирования	9	Ассемблер	1
		Промежуточного уровня	2
		Высокого уровня	3
Тестовое и диагностическое обеспечение	5	Автономное	1
		В составе ОС	2
		Технология изготовления	
Плотность установки ИМС (колич. корпусов/дм^2)	10	Не более 10	3
		10...13	2
		13...16	1
		Более 16	0,5
Тип печатных плат	9	Односторонний (двусторонний печатный монтаж)	3
		Многослойные	1
Реализация управления	8	Схемное	3
		Микропрограммное	1
		Промышленное анонсирование	
Тип производства	10	Серийное	3
		Мелкосерийное	2
		Единичное	1
Степень документирования	9	Полное	3
		Частичное	1
Возможность заказа	8	Свободная	3
		Ограниченная	2
		Целевая	1

Более объективными представляются оценки эффективности архитектуры контроллеров по комплексным критериям, типа производительность/стоимость. В качестве показателя производительности целесообразно использовать Π_c. Стоимость выражается через затраты на реализацию, которая зависит от функциональной сложности контроллера, т. е. количества информации, описывающей таблицу состояний автоматных моделей устройства, значение которой может быть выражено затем через объемы схемотехнического оборудования (например, число корпусов ИС).

Наиболее целесообразный способ оценки эффективности контроллеров в настоящее время — сведение задачи многокритериальной оценки к однокритериальной посредством линейной свертки принятых объективных и субъективных частных критериев.

Весовые коэффициенты, необходимые для вычисления показателя оптимальности, целесообразно определять посредством метода экспертных оценок (табл. 5).

Для детальной оценки каждого показателя используется его уровень относительно показателя «идеального» изделия, соответствующего луч-

шему на данный период времени по рассматриваемой группе показателей.

Метод удобен для получения укрупненной качественной оценки контроллеров при отсутствии методики проведения системометрических измерений технических и программных средств комплексов.

Контроллеры КАМАК для ЭВМ с ОШ

Рациональный ряд серийных контроллеров КАМАК, обеспечивающих радиальное подключение к ядру ИВК, состоит из контроллера крейта и контроллера КПД [4, 5], совместимых между собой конструктивно и по программному каналу. Это обеспечивает многовариантное использование контроллеров в ИВК в зависимости от быстродействия периферийных подсистем.

Контроллеры подобной архитектуры допускают применение нескольких языков программирования: ассемблер; макроассемблер, IML (язык промежуточного уровня); компиляторы, расширяемые драйверами модулей КАМАК (ФОРТРАН); интерпретаторы, также расширяемые драйверами модулей КАМАК (Бейсик, Кейзик, Кэти, Нодал и другие [1]).

Большинство АСНИ организаций АН СССР и АН союзных республик используют ИВК на базе СМ ЭВМ, ЭВМ ряда «Электроника» и средств КАМАК, подключаемых посредством контроллера К-16. Последний программно совместим с контроллером крейта и обеспечивает многовариантность подключения к ОШ различной организации (ОШ СМ, ОШ «Электроника 100/25», ОШ «Электроника 60») и соответствие стандарту МЭК-729, обеспечивающему создание многоконтроллерной периферийной подсистемы на уровне одного крейта [5].

Дальнейшим развитием контроллеров, совместимых по программному каналу с контроллерами типа КК, является целый ряд основных и дополнительных контроллеров, разработанных согласно рекомендациям МЭК-729 [2].

В комплекте с контроллером параллельной ветви, реализованным в виде модуля КАМАК и обеспечивающим возможность одновременного обращения к нескольким крейтам ветви, создаются гибкие и экономичные многокрейтовые системы КАМАК с прямой адресацией к модулям ветви.

Контроллеры крейта реализуются в виде модулей КАМАК шириной 2М. На передней панели располагаются соединители для связи с интерфейсом ЭВМ и разъемы для соответствующих сигналов МЭК-729. На задней панели размещаются два разъема для дополнительной магистрали МЭК-729, для связи с модулем-сортировщиком запросов и организации некоторых связей между контроллером и модулем КПВ.

Многоканальные контроллеры КПД (с числом подканалов до восьми) отличаются существенно более сложной технологией изготовления.

В рамках работ по развитию СМ ЭВМ целесообразна реализация КПМ для интерфейсов ОШ и И41 при условии серийного производства последовательных контроллеров крейта (ПКК) и соответствующей потребности в АСНИ, использующих территориально рассредоточенные подсистемы КАМАК или существенно большое число ППС [5].

При небольшом числе удаленных крейтов достаточно эффективным вариантом является использование и разработка последовательных драйверов и специализированных КК, совместимых с КК и использующих принцип последовательной реализации части протокола ОШ или МПИ по радиальным линиям связи [5].

Конструктивно-элементная база контроллеров СМ ЭВМ второй очереди рассчитана на централизованное серийное производство с использованием обычной технологии. Это накладывает существенное ограничение на реализацию сложных контроллеров КАМАК типа ККПД, ПКК и КПВ в стандарте КАМАК. Ограничения могут быть эффективно преодолены при использовании специализированных для КАМАК, ОШ и И41 микросхем средней и большой степени интеграции, дискретных компонентов, разъемов, переключателей и более совершенной технологии.

Контроллеры МЭК 625-1 для ЭВМ с ОШ

Для небольших АСНИ, не предъявляющих повышенных требований по быстродействию, стандарты системы КАМАК являются избыточными и их применение в ряде случаев экономически нецелесообразно. Для этих целей более приемлем приборный интерфейс, соответствующий стандарту МЭК 625-1 (ГОСТ 26.003-80). Реализация на его основе технических средств более экономична по сравнению со стандартом КАМАК, не требует применения специальных конструкций для модулей, установки устройств и приборов, позволяет использовать для связи устройств достаточно протяженный и быстродействующий 8-разрядный магистральный канал общего пользования (КОП). Алгоритм КОП дает возможность организовать пересылку данных между устройствами различного быстродействия без участия процессора и в различных режимах взаимодействия устройств.

Функционально полное семейство контроллеров КОП должно содержать программный контроллер КПИ, КПИ с каналом прямого доступа, контроллер последовательного КОП.

Параметры контроллеров КОП определяются главным образом способами и степенью реализации интерфейсных функций КОП. Основные Параметры КПИ для ЭВМ с ОШ: схемно-микропрограммная реализация функций источника, приемника и контроллера, за исключением функций дистанционно-местного управления; очистить устройство, параллельный опрос, которые реализуются программным способом. Заложенные в контроллере дополнительные средства позволяют эффективно использовать его в качестве устройства межмашинного обмена, связанного по КОП с аналогичными КПИ других комплексов [3]. Целесообразна реализация КПД в виде одноплатного автономного универсального КПД, архитектура которого не зависит от типа обслуживаемого контроллера ПУ, работающего по программному каналу. Передача данных осуществляется с использованием двух циклов ОШ с быстродействием до 100 Кбайт/с. Автономный универсальный КПД может также эффективно работать с КК, использующимся совместно с КПИ в ИВК на базе ЭВМ с общей шиной.

Контроллер последовательного канала КОП может быть выполнен в виде одноплатного контроллера, программно совместимого с КПИ. Его применение наиболее рационально в низовых комплексах территориально-рассредоточенных ИВК, а также в качестве канала ввода-вывода, обеспечивающего построение локальной вычислительной сети персональных микроЭВМ.

Сформулированные принципы построения системы совместимых контроллеров приборных интерфейсов для СМ ЭВМ могут быть положены в основу аналогичной системы контроллеров для ЭВМ с архитектурой ввода-вывода отличной от ЭВМ с общей шиной.

ЛИТЕРАТУРА

1. Велихов Е. П., Выставкин А. Н. Проблемы развития работ по автоматизации научных исследований. — М.: ИРЭ АН СССР, Препринт № 3 (358), 1980. — 52 с.

2. Структура, технические средства и организация систем автоматизации научных исследований. — Ленинград: ЛИЯФ АН СССР, 1982—172 с.

3. Труды VII Всесоюзной конференции по планированию и автоматизации эксперимента в научных исследованиях. Ч. III. — М.: МЭИ, 1983.

4. Измерительно-вычислительные комплексы для научных исследований. Обзорная информация, вып. 2, ТС-2: ЦНИИТЭИприборостроения, 1980. — 44 с.

5. Мячев А. А., Страшун Ю. П. Современное состояние и тенденции развития системы САМАС. Обзорная информация, вып. 3, ТС-2: ЦНИИТЭИприборостроения, 1983. — 51 с.

Статья поступила 7 сентября 1984 г.

УДК 681.3.06

М. Б. Игнатьев, В. В. Кибиткин, Л. Г. Осовецкий

СТРУКТУРА И ОСОБЕННОСТИ СИСТЕМНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ микроЭВМ

Введение. В настоящее время развитие автоматизированных систем управления (АСУ) любого назначения сопровождается широким внедрением микропроцессорной техники. При этом характерным является в большинстве случаев переход от использования микропроцессорных наборов больших интегральных схем к применению серийных микроЭВМ, которые по своим техническим параметрам не уступают мини-ЭВМ. Этому способствуют возможности аппаратных средств микроЭВМ: появление быстродействующих микропроцессоров с развитой системой команд, микроминиатюрных оперативных запоминающих устройств, малогабаритных внешних запоминающих устройств большой емкости, устройств ввода-вывода широкой номенклатуры.

При построении АСУ на базе микроЭВМ необходима разработка программного обеспечения (ПО). Для этого могут использоваться:

адаптируемые на больших ЭВМ кросс-системы для переноса функционального ПО на микроЭВМ;

унифицированные программы, которые могут включаться как автономные компоненты в ПО различных АСУ на микроЭВМ (далее эти программы будем называть комплектующими программами, а совокупность их — комплектующей системой);

типовые системы программ, обеспечивающие автоматизацию процесса разработки ПО на микроЭВМ (далее эту систему будем называть технологической системой).

Вопросы создания *адаптируемых кросс-систем* достаточно хорошо освещены в литературе [7, 9], поэтому подробнее остановимся на идеологии резидент-систем для проектирования ПО микроЭВМ.

Комплектующая система и *технологическая система* в совокупности составляют системное программное обеспечение (СПО), которое должно рассматриваться как неотъемлемая часть микроЭВМ. В то же время

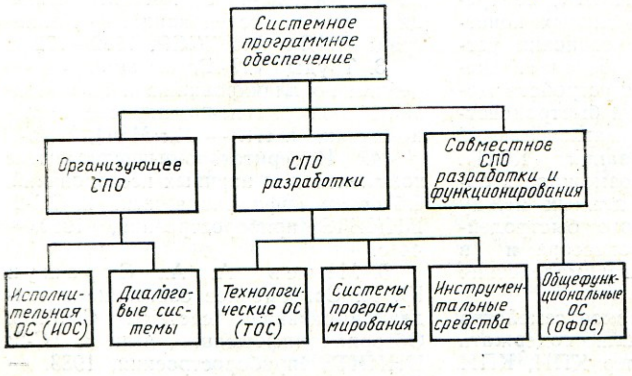


Рис. 1. Взаимосвязь системного и прикладного ПО АСУ

программы, непосредственно выполняющие управление объектами и обработку поступающей информации, составляют функциональное ПО, часть которого полностью зависит от конкретных АСУ. Таким образом, ПО АСУ получается путем объединения прикладного ПО для данной АСУ и некоторых программ комплектующей системы (рис. 1). В этом основная специфика разработки ПО на широко распространенных микроЭВМ в отличие от разработки ПО для специализированных ЭВМ (СЭВМ), по своим техническим характеристикам близким к микроЭВМ, но выпускаемых в небольшом количестве экземпляров. Благодаря архитектурной совместимости и близости систем команд микроЭВМ становится возможным решение проблем мобильности и унификации ПО.

Уровень развития СПО микроЭВМ решающим образом влияет на повышение эффективности и снижение трудоемкости создания ПО АСУ. В то же время трудоемкость разработки СПО и, как следствие, его стоимость чрезвычайно велики. Так, в [1] указывается, что производительность программистов при создании СПО колеблется от 600 до 2500 команд в год, в то время как объем СПО составляет десятки тысяч команд. Поэтому создание и внедрение эффективного и достаточно универсального СПО микроЭВМ в настоящее время представляет собой одну из важнейших проблем, решению которой должно быть уделено первостепенное внимание [2]. Создание СПО, используемое при разработке АСУ, является одной из главных целей научно-технической программы STARS, разработанной в США [3]. Эта программа рассчитана до 1990 г., стоимость ее — 100 млн. долл.

Структура СПО микроЭВМ (рис. 2). При работе микроЭВМ в составе АСУ используются некоторые комплектующие программы из СПО.

Организующее СПО:

программы для организации вычислительного процесса в АСУ в ходе ее функционирования. Совокупность этих программ назовем *исполнительной операционной системой* (ИОС);

программы для обеспечения интерактивного взаимодействия пользователя с АСУ — *диалоговые системы.*

СПО разработки. Режим разработки ПО АСУ обеспечивается и поддерживается технологической системой СПО. Ее основными компонентами являются:

программы для организации вычислительного процесса в ходе разработки ПО АСУ. Совокупность этих программ назовем *технологической операционной системой* (ТОС);

программы для обеспечения разработки ПО АСУ на языках программирования высокого уровня — *системы программирования;*

программы для поддержки и автоматизации отдельных этапов технологического процесса создания ПО — *инструментальные средства.*

Совместное СПО разработки и функционирования. Системы программирования, инст-

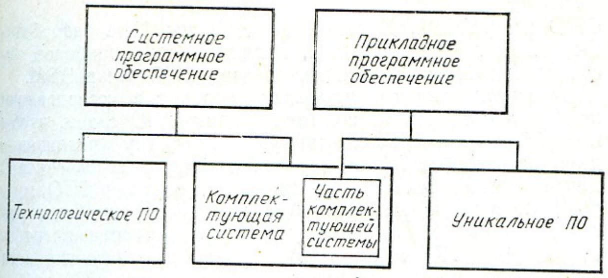


Рис. 2. Структура СПО микроЭВМ

рументальные средства, разрабатываемые прикладные программы и готовое ПО АСУ функционируют совместно. В этом случае для организации вычислительного процесса требуется еще один компонент СПО микроЭВМ — *общефункциональная операционная система* (ОФОС). Поскольку в АСУ используется общефункциональная операционная система, то ОФОС следует отнести к комплектующей системе СПО.

Операционные системы микроЭВМ. Основные требования к СПО микроЭВМ обеспечиваются прежде всего операционными системами (ОС) (см. рис. 2). Мультипрограммный принцип организации вычислительного процесса в микроЭВМ основывается на динамическом перераспределении ресурсов микроЭВМ между несколькими программами. Реализация этого перераспределения и представляет собой основную задачу ОС. Кроме этого, ОС должна решать еще две задачи. Одна из них — управление функционированием внешних устройств микроЭВМ. Решение этой задачи заключается во включении в состав ОС унифицированных программ, управляющих операциями обмена между внешними устройствами и оперативной памятью. Для типовых внешних устройств программы управления обменом должны входить в ОС целиком. Для специализированных внешних устройств (о которых к моменту создания СПО микроЭВМ ничего не известно) должна быть обеспечена возможность по максимальному упрощению написания программ управления обменом с этими устройствами.

Другая (и последняя) задача ОС — обеспечение языкового интерфейса между ОС и программами, выполняемыми в среде ОС (прикладные программы, системы программирования, инструментальные средства), и между ОС и пользователями, работающими за терминалами. Языковый интерфейс служит для формулирования запросов от прикладных программ и пользователей на ресурсы микроЭВМ и на выполнение различных сервисных услуг (копирование данных, находящихся на внешних устройствах, выдача информационных сообщений о ходе вычислительного процесса и т. д.).

Исполнительные операционные системы. Основные функции ИОС: распределение времени процессора в соответствии с фиксированными приоритетами задач; управление функционированием фиксированного (и обычно ограниченного) набора внешних устройств; обеспечение простейшего языкового интерфейса, который включает в себя директивы для взаимодействия задач друг с другом и несколько команд оператора (в основном это команды запуска-завершения задач и команды для получения информации о работе АСУ).

Современные ИОС разделяются на резидентные ИОС (РИОС) и дисковые ИОС (ДИОС). РИОС — это системы, целиком размещаемые в оперативной памяти, и обеспечивающие минимальное время реакции на сигналы со стороны объекта управления. Но объем РИОС очень ограничен и она обычно обеспечивают только вышеперечисленные минимальные функции ИОС. Дальнейшее расширение функцией РИОС производится за счет включения в нее средств динамического распределения оперативной памяти, увеличением возможностей по обеспечению языкового интерфейса при управлении внешними устройствами и данными, команд оператора и т. д. При этом увеличивается объем оперативной памяти, необходимой для РИОС (для современных РИОС требуются от 2,5 до 20 Кбайт).

При расширении функций ИОС на некотором этапе приходится переходить от резидентных систем к дисковым. ИОС размещаются на магнитном диске, с которого их соответствующие компоненты вызываются в оперативную память для выполнения. Для работы ДИОС требуется специальная область оперативной памяти, объем которой не превышает объем, необходимый для РИОС; работают ДИОС значительно медленнее РИОС из-за задержек, связанных с обращениями к магнитному диску.

Технологические операционные системы (ТОС). Основные требования: 1) обеспечение реализации технологического процесса разработки ПО; 2) обеспечение организации многопользовательского режима разработки ПО. Согласно первому требованию ТОС должны обеспечивать управление всеми системами программирования и инструментальными средствами, используемыми при разработке ПО. С технологической точки зрения эти средства мало чем отличаются от прикладных программ.

Второе требование к ТОС определяет: максимальную степень соответствия используемых компонент архитектуры технологической и функциональной микроЭВМ для исключения из технологического цикла кросс-переноса функционального ПО; высокую степень динамичности распределения ресурсов (например, оперативная память распределяется только динамически); широкое использование разнообразных способов виртуализации оперативной памяти; обеспечение средств многопользовательской защиты, которые являются ключевыми при организации режима использования инструментальных средств.

К специфике ТОС относятся: обработка прерываний, происходящих при работе средств динамической отладки программ, средств вывода дампа памяти и трассировки; развитые средства управления данными, что необходимо для эффективного ведения библиотеки поддержки разработки; развитые средства языкового интерфейса в части команд оператора.

Фактическую основу для обеспечения защиты пользователей друг от друга представляет удовлетворение требования к ТОС, состоящего в высокой степени автоматизации распределения ресурсов инструментальных средств. С помощью непосредственных средств многопользовательской защиты в основном контролируется правомерность обращения к ресурсам со стороны конкретного пользователя.

Общефункциональные операционные системы (ОФОС). Их чаще всего можно получить путем выделения некоторого подмножества компонент ТОС.

Инструментальные средства программирования. Они классифицируются в соответствии с принятыми для данного типа АСУ этапами технологического процесса разработки ПО. Так, например, для этапа структурного построения ПО, когда ПО разбиваются на подсистемы, формулируются спецификации и межмодульные интерфейсы для подсистем; необходимым средством является программа, позволяющая производить структурное проектирование.

Для этапов отладки (детерминированной, стохастической, комплексной и т. д.) ПО созданы следующие средства: динамические отладчики, позволяющие в ходе выполнения отлаживаемой программы наблюдать за ее работой; программы вывода дампа памяти и трассировки, выдающие содержимое оперативной памяти в указанные моменты времени или через определенные интервалы времени; генераторы тестов; автоматизированные системы планирования отладки; автоматизированный анализ результатов тестирования программ и др.

Системы программирования для микроЭВМ. В систе-

му программирования входят трансляторы с языков программирования и библиотеки стандартных подпрограмм для этих языков. В соответствии с [4] к языкам программирования относятся структурные ассемблеры и языки высокого уровня (ориентированные на микропроцессоры, проблемно-ориентированные).

Структурные ассемблеры с использованием символических обозначений кодов операций и адресов памяти — это первый шаг в разработке средств автоматизации программирования микроЭВМ. В настоящее время для микроЭВМ подобные языки (ассемблеры) проектируются сразу же с ориентацией на применение методов структурного программирования. Для этого в состав ассемблеров вводятся макроопределения, реализующие основные конструкции структурного программирования. Примером структурного ассемблера является язык PLZ-ASM [4].

Машинно-ориентированные языки в меньшей степени, чем ассемблеры, ориентированы на особенности архитектуры конкретной микроЭВМ, поскольку их основные операторы записываются в форме, близкой к символике обычных алгоритмических языков. Однако допустимые синтаксические формы записи выражений и операндов находятся в прямом соответствии с набором команд и форматом данных конкретного микропроцессора, поэтому каждая микроЭВМ имеет свой машинно-ориентированный язык или диалект некоторого подобного языка. Примерами языков рассматриваемого класса являются Smal/80, Mistral и PL/65.

Языки, ориентированные на микропроцессоры, имеют следующие отличительные особенности:

в них не включены языковые средства, требующие для своей реализации большого расхода аппаратурных ресурсов (прежде всего оперативной памяти);

трансляторы с этих языков совместно со средствами ТОС обеспечивают оптимизацию объектных модулей;

при написании программ можно использовать машинные команды и средства детального управления распределением оперативной памяти;

в них допускаются только те типы данных, которые могут быть реализованы аппаратными средствами микроЭВМ.

Наиболее известными языками высокого уровня, ориентированными на микроЭВМ, являются языки PL/M и PLZ-SYS.

Основные особенности проблемно-ориентированных языков — их стремление к максимально полному обеспечению современных методов технологии программирования и стремление к переносимости программ, написанных на этих языках, с одной микроЭВМ на другую. Наиболее используемые языки для микроЭВМ — С и Паскаль.

Из вышеперечисленных компонент СПО микроЭВМ нерассмотренными остались диалоговые системы. Подробный анализ диалоговых систем дан в [5].

Основные проблемы развития СПО микроЭВМ — организационные, методологические, структурные, технологические, технические и эксплуатационные — непосредственно связаны с общей проблематикой развития программного обеспечения ЭВМ [6].

Организационные проблемы заключаются прежде всего в организации рассмотрения СПО микроЭВМ как самостоятельного промышленного изделия, выпуск которого должен планироваться отдельной строкой в народнохозяйственных планах предприятий-разработчиков и предприятий-изготовителей вычислительной техники и систем управления. Разработка, изготовление, тиражирование и сопровождение СПО микроЭВМ имеют специфические особенности, сильно отличающие их от соответствующих работ по прикладным программам. Эти особенности определяются прежде всего долговечностью СПО микроЭВМ и широкими масштабами его внедрения. Поэтому крайне актуальным является организация разработки и внедрения специальных технологических процессов для создания и сопровождения СПО микроЭВМ. Кроме того, должны быть разработаны специальные методики, определяющие наиболее эффективные способы использования СПО микроЭВМ.

Методологические проблемы состоят в исследовании процесса создания и сопровождения СПО с тем, чтобы выработать формализованную методику реализации данного процесса. Эта методика должна стать основой для создания технологии программирования СПО микроЭВМ. Крайне желательным является разработка количественных методов оценки эффективности разрабатываемого СПО микроЭВМ и оценки эффективности технологии программирования СПО микроЭВМ.

Структурные проблемы связаны с созданием унифицированной структуры СПО микроЭВМ на основе современных методов модульного и структурного программирования. Поскольку функции СПО микроЭВМ строго установлены, то можно добиться высокой степени структурной унификации для различных СПО. Это обеспечит основу для разработки отмеченной выше формализованной методики создания СПО микроЭВМ.

К *технологическим проблемам* относится создание высокоэффективной технологии программирования, обеспечивающей комплексную автоматизацию всех этапов жизненного цикла СПО микроЭВМ. Составной частью технологии должны являться средства автоматизации управления ходом разработки СПО и средства автоматизированного контроля и нормирования труда программистов.

Технические проблемы состоят в разработке методов построения СПО, обеспечивающих удовлетворение современных технических требований к СПО. Для отдельных компонентов СПО микроЭВМ эти требования различаются.

При эксплуатации операционных систем основное требование — возможность адаптации их к условиям применения. Под последними понимается полный комплекс требований к ОС, возникающий в ходе конкретной разработки АСУ: используемые аппаратные средства, набор реализуемых прикладных программ, режим реализации прикладных программ (выполнение или разработка), необходимая степень динамичности распределения ресурсов, критерии распределения ресурсов, удобство использования языкового интерфейса и т. д.

Операционная система должна обладать возможностью как по внешней, так и по внутренней адаптации. Под внешней адаптацией понимается возможность ОС адаптироваться к условиям, которые не предусматривались разработчиком ОС (например, управлять обменом со специализированными внешними устройствами АСУ, разработанными после создания ОС). Очевидно, предполагается обязательное расширение ОС за счет включения в нее новых программ. Для обеспечения внешней адаптации ОС должна быть построена таким образом, чтобы при включении новых программных модулей не пришлось изменять существующие модули ОС, и содержать средства, обеспечивающие легкое включение новых модулей.

Возможны три типа внешней адаптации ОС [7]:

1) надежностная — производится для исправления ошибок, обнаруживаемых в ходе эксплуатации ОС и заключается в переработке ее отдельных модулей;

2) локальная модификация — доработка ОС, производимая пользователем ОС;

3) расширение ОС — это доработка ОС, выполняемая разработчиком ОС.

Четко разделить границы между типами внешней адаптации ОС невозможно, поэтому их часто отличают друг от друга просто объемом вносимых в ОС изменений. Однако с точки зрения построения ОС типы внешней адаптации различаются довольно сильно, поскольку для своей реализации они требуют различных средств.

Под внутренней адаптацией ОС понимается адаптация в рамках, заданных при ее разработке, и поэтому означает выбор некоторого подмножества ОС из общего их множества. Основное средство обеспечения

внутренней адаптации ОС — известный механизм генерации.

Отдельно следует отметить адаптацию ОС к аппаратным средствам, на которых она реализуется, известная под названием мобильность ОС. Решение ее основывается на создании языка программирования, для которого может быть создан высокоэффективный, мобильный транслятор. Например, получил широкое распространение язык С, использованный для разработки операционной системы UNIX.

Основное требование к инструментальным средствам программирования — обеспечение комплексной автоматизации всех этапов разработки программного обеспечения АСУ и взаимной увязки всех инструментальных средств.

При современных технических требованиях язык должен:

поддерживать современные технологические принципы разработки ПО СА, особенно принципы модульности, использования структурированных конструкций, многозадачности;

обеспечивать переносимость прикладных программ между машинами с близкой архитектурой;

обеспечивать написание программ в таком виде, чтобы их можно было бы затем использовать в самых разнообразных системах, т. е. язык программирования должен обеспечивать создание «банков программ».

Эксплуатационные проблемы — проблемы применения ЭВМ на установках пользователя. При всем разнообразии этих проблем основа для их решения — наличие у пользователя комплекса конкретных знаний об используемом СПО. Поэтому следует: разработать унифицированные методы описания СПО, ориентированные на эксплуатацию СПО, т. е. выявляющие те аспекты СПО, которые важны для сравнительного анализа СПО и для его применения; разработать методики обучения пользователей СПО; организовать эффективные формы распространения, сопровождения, рекламирования и обучения СПО.

ЛИТЕРАТУРА

1. Брукс Ф. П. Как проектируются и создаются программные комплексы: Пер. с анг. — М.: Наука, 1979.
2. Мясников В. А. Совершенствование технологии программирования — важнейшая народнохозяйственная задача. — Управляющие системы и машины, 1980, № 1.
3. Martin E. W. Strategy for a DOD software initiative — Computer, 1983, v. 11, N 3.
4. Grespi — Righirri S., Corti P., Dapra A. A survey of microprocessor language — Computer, 1980, v. 8, N 1.
5. Павлюк О. В., Россиков В. В. Диалоговые системы управления и обработки информации: основные требования и их реализация. — Измерения, контроль, автоматизация, 1983, № 3.
6. Липаев В. В. Состояние и проблемы производства программного обеспечения для систем управления и обработки информации. — Управляющие системы и машины, 1980, № 1.
7. Липаев В. В. Качество программного обеспечения. — М.: Финансы и статистика, 1983.
8. Frank G. R. Modular operating systems. — CERN science reports, 1976, N 24.
9. Штрик А. А. Анализ эффективности микропроцессорных комплексов при параллельном исполнении информационно связанных программ. — Управляющие системы и машины, 1983, № 1.

Статья поступила 17 сентября 1984 г.

УДК 681.3.06

А. Б. Борковский

МНОГООКОННОЕ ТЕКСТОВОЕ ВЗАИМОДЕЙСТВИЕ С ПЕРСОНАЛЬНОЙ ЭВМ

Работая на персональной ЭВМ, оператор пользуется различными подсистемами. Для эффективной работы необходимы удобные средства перехода от одной деятельности к другой и обратно и единообразие интерфейса.

При однородном диалоговом взаимодействии пользователя с несколькими работами на экране персональной ЭВМ формируется несколько прямоугольных окон, каждое из которых связано с какой-либо работой. В каждый момент на экране изображено несколько окон, из которых одно — активное и обеспечивает взаимодействие с активной работой. Так как большое число применений ЭВМ можно описать в терминах просмотра и модификации текста, в основу организации взаимодействия положена метафора редактирования.

Концепция многооконного редактирования

При работе с персональной ЭВМ (ПЭВМ) пользователь выполняет различные работы: редактирует тексты, запускает программы, работает с каталогом файлов, строит и отлаживает новые программы. Человек с карандашом и бумагой делает заметки о различных частях работы на листках бумаги и раскладывает их на столе удобным образом. Перед ним лежит несколько листков и раскрытых книг, к которым он может легко обратиться при написании нового листка. Предлагаемый подход к организации взаимодействия с ПЭВМ позволяет работать с машиной в подобном стиле, как бы перенося изображение поверхности стола с заметками и книгами на экран ПЭВМ.

Каждой работе соответствует часть экрана — *окно*, в котором изображаются результаты и сообщения соответствующей программы и в котором пользователь задает команды для этой программы. Одновременно могут быть видны несколько окон, но лишь одно из них — активное. Пользователь взаимодействует с программой, связанной с активным окном, но видит при этом текст и в других окнах. Так, редактируя текст программы в окне редактора, удобно иметь перед глазами результаты ее работы или текст ее спецификаций. Этот текст можно расположить и в другом окне. Работая с каталогом файлов в одном окне, пользователь может просматривать содержимое интересующих его файлов в другом.

Чтобы начать или продолжить работу с другой программой, необходимо обратиться к другому окну. При активизации окна (т. е. при начале или продолжении работы соответствующей программы) оно изображается на экране, возможно перекрывая изображения других окон. Однако содержимое окон не теряется. Будучи повторно активировано, оно как бы всплывает, и его изображение появляется поверх перекрывших его окон. Пользователь может выбирать расположение окон так, чтобы нужная ему информация не исчезала с экрана при переходе от одного окна к другому.

Как организовать работу в одном окне. Одно из основных применений ЭВМ — редактирование текстов.

Пользователи работают с текстами программ, данных и результатов, с текстовыми документами и другой текстовой информацией. Экранный редактор обеспечивает исключительно удобные средства просмотра и модификации текста: можно подгонять курсор к любому месту текста, вставлять и (или) удалять буквы, строки или произвольные фрагменты текста. При этом то, что видно на экране, полностью соответствует содержимому текста. Однако вне экранного редактора эти возможности недоступны: приходится вводить текст команд посимвольно, текст, который уже есть на экране, можно лишь читать, но нельзя использовать как команду для ЭВМ. Фактически, видеотерминал используется как телетайп.

Метафора редактирования. Любую работу предлагается рассматривать как редактирование [7]. Для этого используется следующая модель.

Прикладная программа работает с некоторой структурой, которая может рассматриваться как совокупность однородных объектов. Работа идет одновременно с ограниченным числом объектов, образующих *поле зрения*. Каждый объект имеет представление в виде текстовой строки. Таким объектами могут быть строки текста, для программы работы с каталогом файлов — элементы каталога, состоящие из имени файла и его атрибутов, для СУБД — записи базы данных, для диалогового монитора — пункты меню. Объекты поля зрения представляются в виде текста в окне. Нажимая на клавиши, пользователь воздействует на текст в окне и вызывает действия прикладной программы.

Программа выполняет действия, указываемые пользователем. Часть этих действий универсальна для всех приложений — *перемещение* поля зрения, *выбор* текущего объекта, *модификация, вставка* и *удаление* объектов, а также *выход* из программы и *обращение за подсказкой* (Help).

При переходе от одного приложения к другому смысл «универсальных» операций сохраняется, поэтому они вызываются одинаковым образом во всех программах, и клавиши для них выделены. Конкретный эффект этих операций зависит от приложения, но, с точки зрения пользователя, все происходит однотипно: он видит в текущем окне некоторый текст, по которому можно перемещаться, нажимая на клавиши управления курсором. При достижении границы окна, текст начинает бежать вверх или вниз так, что можно видеть другую часть текста. Если пользователь изменит текст в какой-либо строке, программа отразит изменения в своей внутренней структуре данных и произведет связанные с этим действия. Нажатие клавиши «удалить строку» передает программе запрос об удалении объекта, на текстовое представление которого указывает курсор; вместе с удалением объекта исчезает соответствующая строка окна. Аналогично, клавиша «вставить строку» вызывает новый «пустой» объект в том месте внутренней структуры данных, которое соответствует текущей строке окна.

Для многих приложений лишь часть операций имеет смысл: например, допускается только просмотр или только просмотр и модификация без вставки и удаления. С другой стороны, приложение обычно имеет собственные, неуниверсальные действия. Эти действия вызываются либо как побочный эффект модификации — текст строки может рассматриваться прикладной программой как команда, либо с помощью управляющих клавиш, не имеющих универсального смысла.

Система управления окнами

Система управления окнами (СУОК) организует взаимодействие пользователя и программы на основе метафоры многооконного редактирования.

СУОК позволяет изобразить на экране несколько прямоугольных окон, каждое из которых функционирует независимо в виде отдельного экрана. С каждым окном связана прикладная программа.

При работе СУОК нажатие клавиши считается командой пользователя. СУОК обрабатывает команды клавиш, нажимаемых пользователем, и выполняет операции редактирования в пределах одной строки и перемещения в пределах окна. При нажатии других клавиш СУОК обращается к прикладной программе — хозяину окна, чтобы она помогла выполнить соответствующую команду. При этом СУОК дает прикладной программе средства обеспечить соответствие между ее данными и изображением в окне. Рассмотрим подробнее взаимодействие прикладной программы и СУОК.

Объекты и изображения. Для данной СУОК считается, что прикладная программа (ПП) работает с некоторыми объектами, для каждого из которых определены идентификатор, текстовое представление, предыдущий и следующий объект. Это предположение позволяет СУОК изображать объекты в окне экрана в виде последовательности строк. ПП сообщает СУОК идентификатор объекта и номер строки окна, в которой он должен быть изображен. Запросив у ПП идентификаторы предыдущих и следующих объектов, СУОК заполняют остальные строки окна, а затем запрашивает у ПП текстовые представления объектов и изображает их на экране.

В системах со взаимодействием пользователя с экраном без СУОК программе приходится следить одновременно за двумя структурами данных — за обрабатываемой информацией и за изображением на экране. Их согласование — сложная задача. Оно затрудняет разработку таких программ.

При использовании СУОК ПП работает только с самими объектами и не заботится об устройстве окон. Всякий раз, когда изменяется какой-либо объект или структура данных (т. е. появляются или исчезают объекты), ПП сообщает идентификатор изменившегося объекта СУОК, которая соответственно изменяет изображение. Если пользователь отредактировал изображение какого-либо объекта, СУОК сообщает об этом ПП, передавая ей измененный пользователем текст и идентификатор соответствующего объекта.

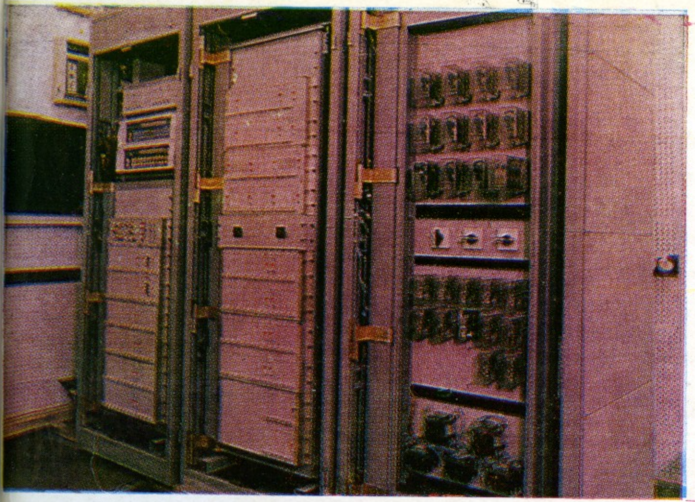
ПП обеспечивает соответствие идентификаторов и объектов, СУОК же никак их не интерпретирует.

Прикладная программа с точки зрения СУОК — экземляр абстрактного типа, для которого определены следующие операции, соответствующие универсальным командам редактирования:

```
ПРИКЛАДНАЯ ПРОГРАММА  =TYPE WITH
СЛЕДУЮЩИЙ: FUNCTION (идент) → идент;
ПРЕДЫДУЩИЙ: FUNCTION (идент) → идент;
ИЗОБРАЖЕНИЕ: FUNCTION (идент) → строка;
ОБНОВИТЬ_ОБЪЕКТ: FUNCTION (идент строка);
ОСОБОЕ_ДЕЙСТВИЕ: PROCEDURE (текущий: идент;
кл: клавиша → (выйти_из_окна: boolean; перейти_к_объекту : идент);
ПОДСКАЗКА: PROCEDURE ( );
```

При создании окна с ним связывается какая-либо ПРИКЛАДНАЯ ПРОГРАММА (ПП). Окно определяет поле зрения пользователя: пользователь может непосредственно обращаться к объектам, изображения которых он видит в окне.

П е р е м е щ е н и е. Команды перемещения задаются клавишами управления курсором. Пока перемещение происходит в пределах окна, СУОК не обращается к ПП. Команда, пытающаяся вывести курсор за верхнюю

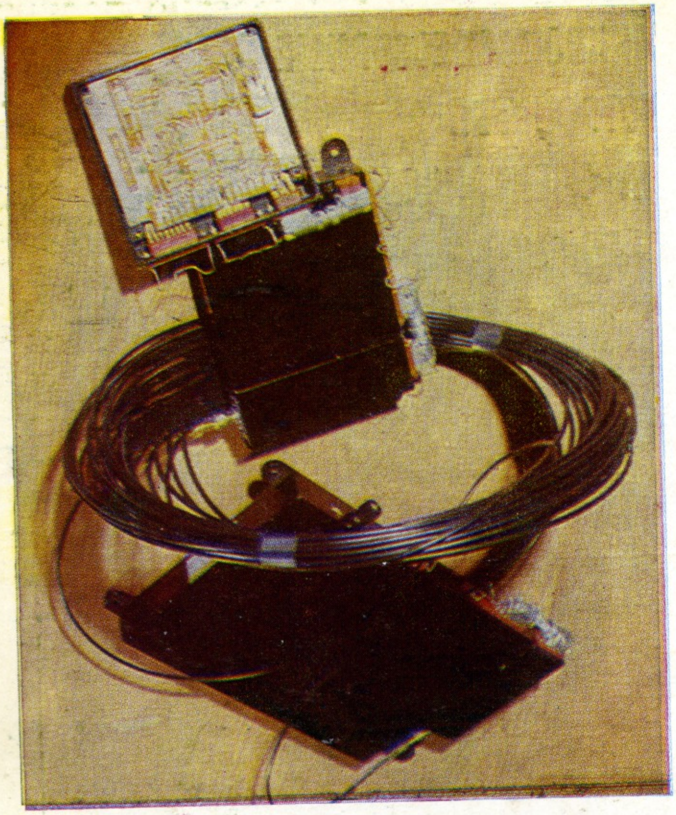


МИКРОПРОЦЕССОРНЫЕ УСТРОЙСТВА НА ЭЛЕКТРИЧЕСКИХ СТАНЦИЯХ

Устройство управления мощностью крупных паровых турбин на основе МСУВТ В7 (к ст. Н. В. Колобродова, Ю. В. Чугунникова и др.)



ОПТОВОЛОКОННЫЕ СИСТЕМЫ



Система сбора аналоговых данных

«ЭЛЕКТРОНИКА МС 8201», «ЭЛЕКТРОНИКА МС 8401»

ОПТОВОЛОКОННЫЕ СИСТЕМЫ СБОРА И РАСПРЕДЕЛЕНИЯ АНАЛОГОВЫХ ДАННЫХ

Оптоволоконная система сбора данных «Электроника МС 8201» предназначена для многоканального преобразования входных аналоговых сигналов в последовательный формат данных, передачи их по оптоволоконному линейному тракту (ОВЛТ) и выдачи данных в виде параллельного формата на выходе системы.

Оптоволоконная система распределения данных «Электроника МС 8401» преобразовывает параллельные форматы данных в последовательный формат, передает их по ОВЛТ, а также осуществляет многоканальное преобразование данных в аналоговые сигналы на выходе системы.

Рекомендуются для применения в комплексах связи и управления, комплексах межмашинного обмена, АСУТП, вычислительных системах с развитой периферией, измерительных комплексах, автоматизированных системах управления и контроля.

Техническая характеристика

	«Электроника МС 8201»	«Электроника МС 8401»
Диапазон входных аналоговых сигналов, В	±10	—
Диапазон выходных аналоговых сигналов, В	—	±10
Относительная приведенная погрешность преобразования аналогового сигнала, %, не хуже	0,1	0,1
Число аналоговых дифференциальных входов	16	—
Число аналоговых выходов	—	8
Число параллельных цифровых входов	—	16
Число параллельных цифровых выходов	16	—
Уровни входных—выходных цифровых сигналов	ТТЛ	ТТЛ
Разрешающая способность трактов АЦП и ЦАП	12 двоичных разрядов в параллельном двоичном коде	
Полоса частот преобразования сигнала тракта, Гц:		
АЦП	0—700	—
ЦАП	—	0—15000
Скорость передачи цифровой информации в трактах, Мбит/с	8	8
Напряжение питания, В	+15+15%	+5+5%
Потребляемая мощность, Вт, не более	10	10
Диапазон рабочих температур, °С	—10...+55	
Габаритные размеры каждого блока, мм, не более	145×205×20	
Длина оптического кабеля, м, не более	300	300

Годовой экономический эффект от внедрения 100 систем составляет 3717,8 тыс. руб.





Набор инструмента и оснастки для монтажа и прокладки оптоволоконных трактов

ПЕРЕДАЧИ ДАННЫХ

«ЭЛЕКТРОНИКА МС 4101»

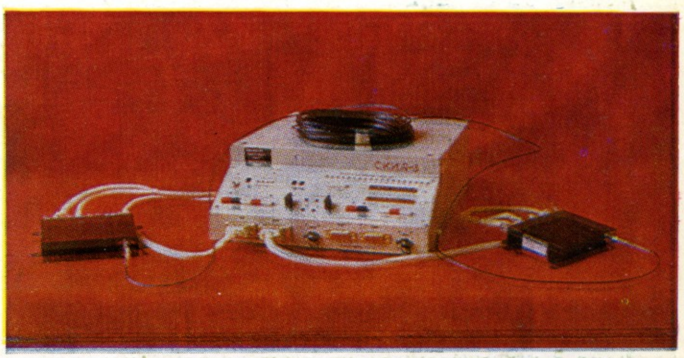


Цифровая система передачи данных

МУЛЬТИПЛЕКСИРОВАННАЯ ОПТОВОЛОКОННАЯ ЦИФРОВАЯ СИСТЕМА ПЕРЕДАЧИ ДАННЫХ

Предназначена для преобразования параллельных форматов данных в последовательный формат, передачи их по оптоволоконному тракту и обратного преобразования — последовательного формата данных в параллельные.

Рекомендуется для применения в информационных комплексах машинного обмена, АСУТП, вычислительных системах с развитой периферией, автоматизированных системах управления и контроля.



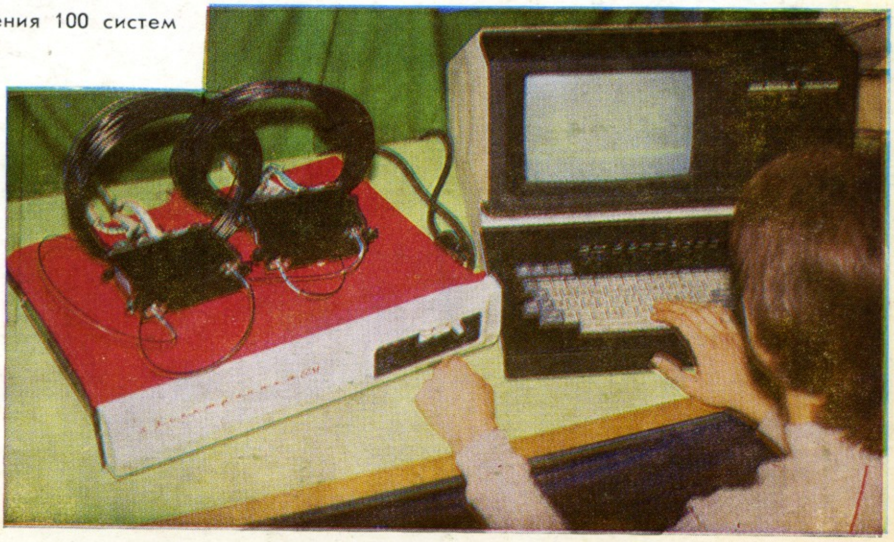
Стенд цехового автоматизированного контроля изделия «Электроника МС 4101»

Техническая характеристика

Число параллельных входов—выходов	19
Скорость передачи информации, Мбит/с	8
Уровни входных — выходных сигналов	ТТЛ
Вероятность ошибки в линейном тракте	10^{-9}
Напряжение питания, В	±5 ±5%; ±10 ±5%
Потребляемая мощность, Вт, не более	8
Диапазон рабочих температур, °С	—10...+55
Длина оптического кабеля, М, не более	300
Габаритные размеры, мм, не более	160×110×22

Система имеет повышенную защищенность от внешних электромагнитных полей, не создает электромагнитных помех, допускает высокоскоростную передачу информации, имеет повышенную надежность функционирования в агрессивных, взрыво- и пожароопасных средах, значительно сокращает массогабаритные показатели трактов передачи информации, упрощает прокладку линий связи на действующих объектах.

Годовой экономический эффект от внедрения 100 систем составляет 2595 тыс. руб.



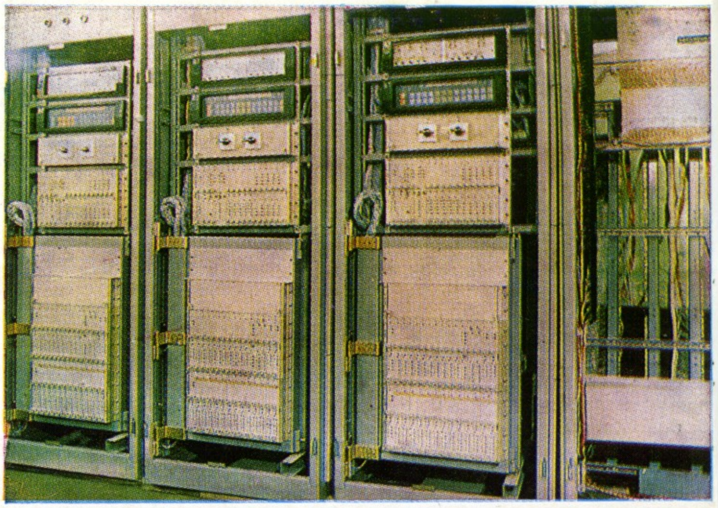
Оптоволоконная цифровая система передачи данных — средство связи микроЭВМ с дисплеем

Адрес для запроса документации и справок: 117415, Москва, В-415, ЦНИИ «Электроника».

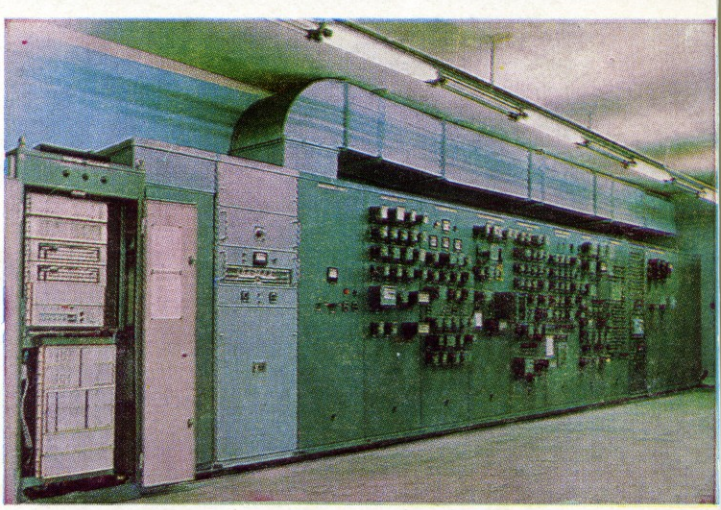
Удовлетворение заявок на техническую документацию производится до 31.12.88 г.

МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА УПРАВЛЕНИЯ РЕЖИМАМИ ЭЛЕКТРОСТАНЦИЙ

(к ст. В. Д. КОВАЛЕВА,
А. К. ХАНСУВАРОВА,
А. Т. ШЕВЧЕНКО)

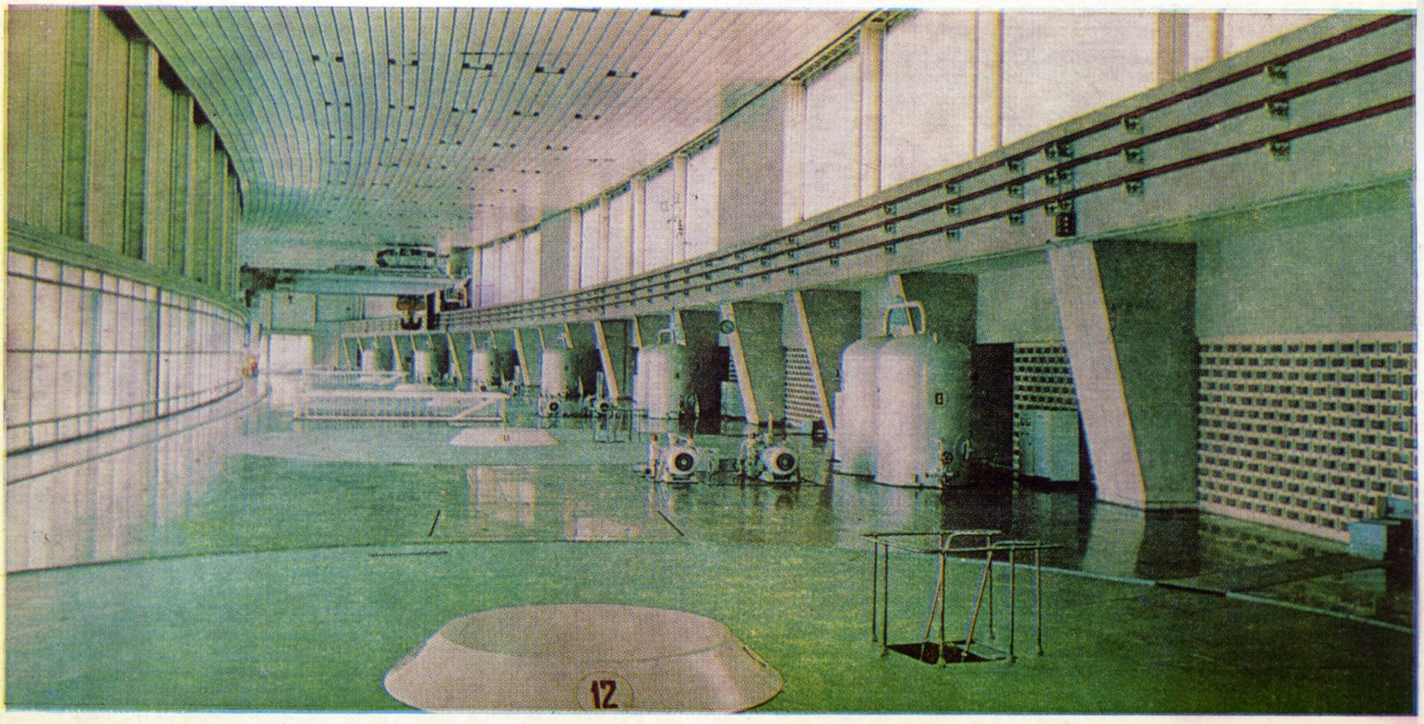


Система противоаварийной автоматики на базе МСУВТ В7 вычисляет управляющие воздействия (УВ) для сохранения устойчивости электростанций при аварийных возмущениях и распределяет вычисленные и полученные УВ от устройств противоаварийного управления высшего уровня между агрегатами электростанции



Стойка управления возбуждением синхронного генератора в зависимости от напряжения сети ГЭС

Общий вид машинного зала



или нижнюю границу окна, понимается как требование пользователя переместить поле зрения. Однако СУОК ничего «не знает» о том, что не изображено в окне. Поэтому она обращается к функциям ПП.СЛЕДУЮЩИЙ и ПП.ПРЕДЫДУЩИЙ за идентификаторами объектов, изображения которых должны появиться в окне, а затем к функции ПП.ИЗОБРАЖЕНИЕ, которая выдает текст, соответствующий этим объектам. Если объект не имеет следующего (предыдущего), то окно не изменяется.

Соответствие изображения и данных ПП. В результате редактирования в пределах строки (при изменении строки в окне) изображение в окне перестает соответствовать структуре данных ПП. Изменение изображения опережает изменение данных. Поэтому, всякий раз, когда пользователь перемещает курсор из строки, СУОК проверяет, изменял ли он эту строку. Если да, то измененная строка передается прикладной программе посредством процедуры П.ОБНОВИТЬ ОБЪЕКТ (она анализирует внесенные пользователем изменения и выполняет соответствующие действия). В частности, если изменения недопустимы с точки зрения программы, процедура может сообщить об этом пользователю и восстановить изображение в окне.

Действия, зависящие от приложения. Для более эффективного общения удобны проблемно-ориентированные одноклавишные команды. Поэтому при нажатии управляющей клавиши, не имеющей универсального смысла, СУОК ничего не делает сама, и только обращение к ПП сообщает ПП, что была нажата клавиша К:ОСОБОЕ–ДЕЙСТВИЕ (Текущий–объект, К). ПП выполняет необходимые действия, сообщает СУОК об изменениях, которые произошли при этом с объектами. Функция ОСОБОЕ_ДЕЙСТВИЕ возвращает СУОК два параметра: первый (boolean) указывает на конец работы с данным окном; второй — куда переместилось поле зрения и какой объект должен стать текущим; СУОК при этом действует так же, как при перемещении курсора.

Подсказка. Общепризнано, что диалоговая система должна уметь объяснить пользователю, где он находится, что происходит, и что, и как он может сделать. Запрос о помощи должен производиться одинаково во всех приложениях. Поэтому для СУОК выделяется универсальная управляющая клавиша запроса о подсказке. При нажатии этой клавиши СУОК вызывает ПП.ПОДСКАЗКА. В состав СУОК входит программа просмотра текста подсказки в выделенном для этого окне, и разработчику прикладной программы достаточно лишь указать файл с соответствующим текстом. Однако он может задать и собственный способ управления диалоговой документацией.

Выход. Универсальной командой является также выход из окна.

Ниже, при описании конкретной реализации, приведены примеры конкретных ПРИКЛАДНЫХ ПРОГРАММ и их операций.

СУОК с точки зрения программы. В предыдущем разделе описано, как работает СУОК, если вся работа с окном управляется одноклавишными командами пользователя. Однако изменения в окне могут быть произведены и программой. Для этого СУОК определяет окно как абстрактный тип с операциями, позволяющими управлять изображением в окне:

```
Окно=TYPE WITH
Высота, Ширина, Нач–строка, Нач– столбец...:число;
СОЗДАТЬ– ОКНО:  FUNCTION (ПП:ПРИКЛАДНАЯ –
                   ПРОГРАММА;
                   xyz:параметры окна) → Окно;
ОТКРЫТЬ– ОКНО:  PROCEDURE (О: ОКНО);
ИЗМЕНЕН:        PROCEDURE (и: идент);
ВСТАВЛЕНЫ:      PROCEDURE (перед: идент; сколько:
                   число);
УДАЛЕНЫ:        PROCEDURE (начиная–с: идент; сколь-
                   ко: число);
ДИАЛОГ:         PROCEDURE (  );
ЗАКРЫТЬ ОКНО: PROCEDURE (  );
```

Операция СОЗДАТЬ–ОКНО создает новое окно как структуру данных соответственно указанным параметрам (размеры и положение окна, цветовые атрибуты, режим редактирования и пр.). Со вновь созданным окном связывается указанная ПП. При этом изображение на экране не изменяется.

Операция ОТКРЫТЬ–ОКНО высвечивает на экране текст указанного окна и делает его *текущим окном*. Все остальные операции производятся с текущим окном.

Мы уже отмечали, что прикладная программа не работает непосредственно с изображением. Операции ВСТАВЛЕНЫ, УДАЛЕН и ИЗМЕНЕН указывают не место на экране, а идентификатор объекта, с которым связаны изменения. Прикладной программе не надо «знать», в какой строке окна расположен объект; более того, ей не надо «беспокоиться» даже о том, изображен ли он вообще на экране. Операция ВСТАВЛЕНЫ, УДАЛЕН и ИЗМЕНЕН — *информационные;* они не предписывают СУОК конкретных действий, лишь извещают о произошедших изменениях. Когда нужно, СУОК оптимальным способом приводит изображение на экране в соответствие с данными прикладной программы.

Операция ДИАЛОГ делает текущее окно активным, т. е. переводит его в описанный в предыдущем разделе режим управления одноклавишными командами пользователя.

При операции ЗАКРЫТЬ–ОКНО изображения окон, перекрытых текущим окном, «всплывают» и перекрывают его изображения. Окно временно уходит из-под контроля СУОК, а текущим становится то, которое было текущим до открытия данного.

Реализация СУОК

Вариант СУОК был реализован для операционной системы MS DOS [2]. На основе этой СУОК созданы: многооконный редактор системы подготовки текстов АБВ, экранный интерфейс для работы с каталогами файлов и программа редактирования диска в терминах физических единиц обмена. СУОК также организует адаптивный диалог [3] с помощью сети меню в различных прикладных программах.

Многооконный экранный редактор АБВ — первая система, реализованная с помощью СУОК. Редактор позволяет работать в каждом из имеющихся на экране нескольких окон с различными кусками одного файла или с разными файлами.

Размеры и расположение окон редактирования устанавливаются пользователем. При работе с одним окном редактор АБВ обеспечивает обычные средства экранного редактирования: перемещение по тексту с помощью клавиш управления курсором на одну или несколько литер, строк или страниц, перемещение по лексемам, контекстный поиск и замену, вставку и удаление литер, строк и произвольных фрагментов текста, восстановление удаленного текста, копирование фрагмента текста и пр.

Объекты, с которыми работает редактор, — это строки, организованные в двунаправленный список. Идентификаторами служат адреса строк в буфере. Внутреннее представление объектов АБВ совпадает с их изображением в окне, поэтому операции взаимодействия АБВ с СУОК очень просты:

АБВ.СЛЕДУЮЩИЙ и АБВ.ПРЕДЫДУЩИЙ возвращают указатели, связывающие строки в список. АБВ.СЛЕДУЮЩИЙ от последней строки и АБВ.ПРЕДЫДУЩИЙ от первой строки возвращают нулевой идентификатор.

АБВ.ОБНОВИТЬ_ОБЪЕКТ берет из окна измененную строку и помещает ее в буфер без преобразований.

АБВ.ИЗОБРАЖЕНИЕ пересылает в окне необходимые строки буфера.

Посредством АБВ.ОСОБОЕ_ДЕЙСТВИЕ вызываются специальные операции редактирования (вставка и удаление строк, поиск, замена, копирование, перемещение между окнами и др.).

Работа редактора управляется с помощью меню. Для этого, кроме окон редактирования, есть два окна — для меню и для задания параметров.

В окне меню пользователь выбирает, что делать дальше: сохранить файл на диске, загрузить другой файл и т. п. Для указания выбранного действия пользователь переводит курсор в нужную строку экрана и нажимает клавишу «Выполнить». Для меню идентификаторы соответствуют номерам пунктов меню. Функция МЕНЮ.ИЗОБРАЖЕНИЕ выдает текст описания пункта. Процедура МЕНЮ.ОБНОВИТЬ_ОБЪЕКТ ничего не делает, а процедура МЕНЮ.ОСОБОЕ_ДЕЙСТВИЕ выполняет выбранную команду, которая указывается ее первым аргументом (идентификатором текущего объекта). Операция МЕНЮ.ОСОБОЕ_ДЕЙСТВИЕ выполняет действие, соответствующее текущей строке меню.

Окно настройки используется для задания параметров окон, имен файлов и пр. Это окно также является меню, но можно не только выбирать в нем пункт меню, но и заносить в него числовые и текстовые значения. При этом используется способность СУОК защищать части строк от изменений.

Программа работы с файлами (РФ) позволяет просматривать в окне экрана каталог диска и удалять, переименовывать и копировать файлы. Имеется возможность указать шаблон для имен, а не работать со всеми файлами. Выбранный файл можно также просмотреть в другом окне или редактировать его с помощью редактора, описанного выше.

В отличие от редактора, программа работы с файлами хранит в буфере дескрипторы файлов, а не текст. Текст для изображения в окне синтезируется при каждом обращении к РФ.ИЗОБРАЖЕНИЕ. РФ.ОБНОВИТЬ ОБЪЕКТ также разбирает строку, полученную из окна и, возможно, переименовывает соответствующий файл или меняет его атрибуты. Удаление строки влечет за собой удаление соответствующего файла, вставка строки — создание нового файла. РФ позволяет перемещаться по древовидным каталогам MS DOS.

Редактор диска — это системная программа, позволяющая считывать и редактировать содержимое диска или файла поблочно. Она представляет собой экранный дамп. Эта программа во многом сходна с текстовым редактором. Два важнейших отличия: так как измененные блоки записываются на то же место, откуда они были считаны, нет возможности вставлять и удалять символы и строки — все изменения происходят «на месте»; содержимое буфера может изображаться в различных форматах: в символьном, шестнадцатеричном, десятичном, по словам или по байтам и пр. (пользователь может выбрать формат для каждого окна). Функция ИЗОБРАЖЕНИЕ синтезирует изображение в соответствии с выбранным форматом. Кроме текста дампа в выбранном формате, в начале каждой строки изображается адрес ее первого байта, который служит идентификатором объекта.

Опыт реализации и использования

Опыт работы с СУОК показал, что метафора многооконного редактирования удобна для многих приложений. Имеет смысл включить механизм СУОК в операционную систему наравне со средствами работы с файлами. Базовые операции СУОК могут быть реализованы на уровне дисплейного адаптера.

Уровень языка. Реализация СУОК показала, что для системного программирования необходимо использовать язык более высокого уровня, чем Паскаль.

Описанная СУОК реализована на Паскале. Использование языка высокого уровня не уменьшило эффективности и компактности, однако выразительные средства Паскаля не удовлетворяют потребностям данной задачи. Базовые понятия СУОК (связывание объекта и группы процедур) невыразимы средствами Паскаля. Например, описанную в разделе 2 функцию СОЗДАТЬ ОКНО (ПП, xyz) → окно невозможно написать на Паскале. В сущности, в основе СУОК лежит объектно-ориентированный подход [4—6]. Опыт ее разработки выявил ряд преимуществ объектного программирования над процедурным. Для создания развитых систем подобного рода необходима инструментальная система, поддерживающая концепцию объектного программирования.

Направления развития. Набор «универсальных» операций, предложенный в данной статье, полон. Однако ряд операций также можно рассматривать как универсальные: поиск, выделение, копирование. Следует развить СУОК в этом направлении.

Основная концептуальная единица СУОК — окно, состоящее из равноправных строк. В определенных случаях удобно разбить окно на более мелкие самостоятельные единицы — «форточки», каждая из которых по-своему реагирует на операции редактирования.

Изложенный в данной статье подход к организации взаимодействия с ПЭВМ в настоящее время применяется все шире и шире. Метафора редактирования рассматривается в работе [7]. В ряде ПЭВМ текстовые окна предлагаются как основа диалога со всей системой — это, например, ПЭВМ LISA и Macintosh фирмы Apple, новая портативная ПЭВМ фирмы Gavilan, которая, по мнению экспертов, опередила современный уровень программирования по меньшей мере на год [5]. Аппарат окон встроен в систему программирования Smalltalk [4], в различные Лисп-машины, например, в Symbolics-3600 фирмы Symbolics [1]. С распространением 16-разрядных ПЭВМ диалог, основанный на многооконном редактировании, становится одним из основных способов взаимодействия с ПЭВМ.

ЛИТЕРАТУРА

1. «Symbolics 3600 Data Processing Systems. General Overview». — Symbolics, 1982.
2. Larson Ch. MS—DOS Z.O: An Enhanced 16-Bit Operating System. — Byte, 1983, v. 8, No 11, p. 285—289.
3. Брябрин В. М. Адаптивный диалог — основа персональной вычислительной системы. — В кн.: Лингвистические процессоры и представление знаний. Новосибирск, 1981.
4. Goldberg A., Ingals D., Robson D. Smalltalk-80, the Language and its Implementation. — Byte, 1981, v. 6, No 8.
5. Galivan Portable Computer. — Byte, 1983, v. 8, No 6.
6. Liskov B. et al. Abstraction Mechanism in CLU. — CACM, 1977, v. 20, No 8.
7. Fraser C. W. A Generalized Text Editor. — CACM, 1980, v. 23, No 1, p. 22—60.

Статья поступила 2 августа 1984 г.

ТРИБУНА УЧЕНОГО

От редакции. — В этой новой рубрике журнала мы будем знакомить наших читателей с выдающимися учеными нашего времени, вносящими значительный вклад в формирование научных основ и развитие вычислительного дела. Без сомнения, многие из нас знают их имена, штудировали их труды, применяют введенные ими понятия или разработанные ими алгоритмы. Страницы журнала предоставляют, однако, дополнительную благоприятную возможность более прямого контакта с этими учеными путем публикации их выступлений по актуальным вопросам информатики и вычислительной техники, адресованных широкой аудитории. Эти выступления будут дополняться кратким очерком жизненного пути ученого и характеристикой его основных работ. Естественно, что публикуемые в этой рубрике материалы приглашенных авторов выражают их личную точку зрения, не обязательно совпадающую с мнением редакции.

Первый гость нашей рубрики — профессор вычислений Оксфордского университета, член Королевского общества, лауреат премии имени Тьюринга Ч. А. Р. Хоар.

Чарлз Антони Ричард Хоар родился 11 января 1934 г. на Цейлоне в семье колониального чиновника. Его отец, государственный служащий, получил образование в Оксфорде, мать была дочерью чайного плантатора. В 1941 г. во время наступления японских войск в Юго-Восточной Азии семье пришлось на время эвакуироваться в Южную Родезию, пережив опасное морское путешествие по Индийскому океану. Вскоре после войны семья вернулась в Англию и поселилась неподалеку от Оксфорда, навсегда втянувшего Хоара в орбиту своего интеллектуального притяжения.

Учение молодого Хоара развивалось по сложившейся схеме английского классического образования: начальная — подготовительная — публичная школы — Мертоновский колледж Оксфордского университета по отделению литературы и гуманитарных наук (в сопоставлении с нашими университетами — объединение филологического и философского факультетов). Отношение к учебе, по собственному признанию, было довольно формальное, но логика его интересовала, и Хоар активно работал в вечернем студенческом семинаре, штудируя «Математическую логику» У. Куайна.

Вспоминая свой период классического образования, Хоар отмечает, что теперь программирование во многом заменяет латынь как средство выработки привычки к точности и формализму с утратой, однако, доли гуманитарности, присущей этому древнему языку.

Получив в 1956 г. первую степень бакалавра искусств, Хоар был призван на действительную службу в военно-морской флот.

По окончании службы в 1958 г. Хоар продолжает обучение в Оксфорде, записавшись на выпускной курс по статистике, которая в его представлении как-то интегрировала интересы к философии, логике и вероятности. Через год, благодаря знанию русского языка, Хоар был по обмену направлен в СССР в Московский университет на годичную стажировку по математической статистике на кафедре теории вероятности. С русским языком все оказалось в порядке, но для продуктивной работы в семинаре доцента Б. А. Севастьянова Хоару не хватило математических знаний и поэтому «главное дело» не пошло. Тем не менее обстановка в Москве была достаточно стимулирующей, чтобы дать молодому человеку ряд импульсов, повлиявших на его дальнейший жизненный путь.

Национальная физическая лаборатория, один из ведущих центров развития и применения вычислительной техники в Англии, предполагала начать исследования по машинному переводу и обратилась к Хоару с предложением в порядке подготовки изучить советские работы в этой области. Он взялся за дело, регулярно посещая активные в то время московские семинары по машинному переводу. К этому периоду относится первая научная публикация Хоара. Это была статья на русском языке, посвященная синтагматическому анализу синтаксических структур и опубликованная в сборнике «Машинный перевод» [1]. Планам работы в Национальной физической лаборатории не суждено было сбыться. Однако именно это исследование алгоритмов анализа на деревьях синтаксического разбора привело Хоара к его известному алгоритму сортировки Quicksort, ставшему для него впоследствии постоянным примером для отработки многих идей, касающихся методов описания языков, анализа программ и доказательства их свойств [2, 3].

Другим внешним событием стала обращенная к Хоару просьба британской Ассоциации научных машин и приборов выступить в качестве гида на выставке приборов, организованной в Москве в 1959 г. Эта принятая им просьба не только обеспечила Хоару увлекательный и бесплатный проезд домой на выставочном автомобиле через всю Европу, но и позволила ему более подробно познакомиться с компанией Эллиот Бразерз, которая, приступая к разработке ЭВМ Эллиот-803, предложила Хоару место программиста в своем отделении вычислительной техники.

Работа в компании началась с разработки библиотеки стандартных подпрограмм — обычный дебют системных программистов 50-х годов. Решающее значение для определения дальнейшей работы сыграло знакомство с Алголом 60. Процедурные свойства языка и способность к выражению рекурсии побудили Хоара к осознанию, одновременно, необходимости и возможности разработки научных основ программирования. Это позитивное влияние Алгола 60 было подкреплено успехом разработки транслятора с этого языка для Эллиот-503, ставшего одной из первых производственных реализаций языка [4].

В результате этого первого успеха в 1963 г. Хоар увидел себя в качестве руководителя большого коллектива программистов, разрабатывающего систему Марк-2 программного обеспечения для ЭВМ Эллиот-503 (операционная система, архив и трансляторы с ассемблера, Фортрана и Алгола). Судьба этого проекта стала, можно сказать, классическим выражением судьбы «второй системы» по Бруксу. Система не появилась ни через запланированные полтора, ни через два года. Прошло еще два года авральной работы, прежде чем вместо интегрированной системы пользователи получили скромные по возможностям, но адаптированные к их нуждам комплекты программного обеспечения. Сам Хоар подверг опыт этого предприятия беспощадному и поучительному анализу в своей Тьюринговой лекции в 1980 г. [5].

Уроки этого несостоявшегося проекта самим Хоаром были усвоены, однако, значительно раньше 1980 г. В тот же отрезок 60-х годов Хоар принимал активное участие в работе над новым алгоритмическим языком в составе Рабочей группы Международной федерации по обработке информации (ИФИП). В поисках экономного, элегантного и точного языка он с группой единомышленников, включавшей Никлауса Вирта, завершил работу по уточнению Алгола 60, получившую впоследствии название «ИФИП-овского подмножества Алгола», а затем принял участие в спецификации нового варианта языка, ставшего известным под названием «Алгол W» [6]. В то же время в Рабочей группе ИФИП стал набирать силу подход к разработке большого, богатого понятиями и изобразительными средствами языка. Хоар подверг этот подход критике, в значительной степени опираясь на свой болезненный опыт реализации

полного Алгола 60 и предупреждая о трудностях с формализацией, точности и надежности языка, большого по объему и избыточного по конструкциям. Сторонники этих двух тенденций не смогли выработать общей платформы: большинство Рабочей группы пошло вслед за А. ван Вейнгаарденом к Алголу 68, а меньшинство, включавшее Хоара и Вирта, вышло из группы, сохранив, однако, выбранное направление в проектировании языков, воплотившееся впоследствии в языке «Паскаль». Хоару «Паскаль» обязан оператором CASE. Кроме того, паскалевские записи в значительной степени опирались на ранее проведенное Хоаром исследование структурных данных в языках программирования, опубликованное в его известной статье «Работа с записями» [7].

Корни другого цикла исследований Хоара, принесших ему мировую известность, лежат в реплике, брошенной ему в 1963 г. одним из руководителей компании во время очередного обсуждения плачевного состояния дел с системой Марк-2: «Знаете ли вы, в чем беда? Вы позволяете вашим программистам работать над заданием, которого вы сами не понимаете!» В то время, как вспоминает сам Хоар, эта реплика показалась ему идеалистическим пожеланием постороннего человека, не представляющего реальной сложности и непознаваемости каждым отдельным человеком больших программных комплексов. С годами, однако, особенно в процессе работы над проектированием языков программирования Хоар начал глубже понимать необходимость выработки точных и формальных методов описания языков программирования и самих программ. К тому времени стали появляться механизмы формального описания языков на основе абстрактной машины, которая, в свою очередь, опиралась или на механизм детерминированного исполнения (операционная семантика), или на рекурсивную запись функциональных зависимостей, реализуемых языком (денотационная семантика). Хоара отпугивала громоздкость этих механизмов, их переопределенность деталями представления и другими привходящими обстоятельствами. Главное же — эти механизмы были недостаточно расчленены, они обслуживали язык в целом и не давали прямого способа систематического применения воплощенного в них знания о языке к отдельной программе, например, чтобы доказать те или иные ее свойства.

Углубившись в эти размышления, Хоар понял, что исследования подобного рода требуют иной обстановки, и поэтому вскоре после завершения работ над программным обеспечением ЭВМ Эллиот-503 перешел на преподавательскую работу в Королевский университет Белфаста, где с 1968 г. принял кафедру вычислительных наук. Толчком к решению проблемы послужила работа Роберта Флойда «Описание смысла программы», показавшего, как, начиная с некоторых утверждений, соотнесенных отдельным элементам программы, получить закономерно некоторое суждение, касающееся программы в целом. Ровно через год появилась работа Хоара «Аксиоматический подход к программированию для ЭВМ», заложившая основы доказательного программирования [8].

Аксиоматический подход к описанию моделей вычислений и доказательству свойств в программах начал разрабатываться десятками специалистов во всем мире настолько интенсивно, что Хоар почувствовал возможность переключиться на другой вопрос, поставленный тем же периодом его практической работы в 60-е годы: «Почему так трудно программировать операционные системы?». Систематическую разработку этого вопроса Хоар начал в Белфасте, хотя идея того, что ключ к пониманию структуры операционной системы лежит в дисциплине работы с общей памятью, восходила к более раннему времени и даже обсуждалась с Э. Дейкстрой, но не встретила в тот момент его понимания. Первым существенным результатом оказалось нахождение монитора — языковой конструкции, позволяющей описывать и структурировать операционные системы средствами алгоритмических языков [9]. Идея монитора тоже оказалась весьма плодотворной, была подхвачена многими специалистами и, в особенности, Бринком Хансеном, создавшим «Параллельный Паскаль» и многие его реализации.

В 1977 г. Хоар получил кафедру профессора вычислений в родном Оксфорде, ставшую вакантной в результате безвременной кончины профессора Кристофера Стрейчи, одного из основоположников вычислительного дела в Англии. Избрание Хоара по результатам трудного конкурса утвердило его репутацию ведущего специалиста в области программирования. Здесь получает дальнейшее развитие его работа по параллельному программированию, получившая название теории взаимодействующих последовательных процессов. Эта теория, восходящая к одноименной ранней работе Эдсгера Дейкстры, неузнаваемо преобразилась в результате исследований Хоара, использовавшего аксиоматический метод и алгебраические свойства процессов обработки информации и историй (протоколов) их развертывания во времени [10, 11]. В этом году Хоаром закончена монография, обобщающая почти десятилетний период напряженной работы [12].

Работая большую часть времени в математической теории программирования, Хоар держит постоянно под наблюдением развитие программирования как практической деятельности. Немало его выступлений посвящено трудной проблеме объединения математического и инженерного начал в программировании. Он продолжает консультировать ряд прикладных разработок. В частности, вызвавший интерес у специалистов язык параллельного программирования Оккам в значительной мере разработан под влиянием Хоара.

Для лучших работ Хоара характерно острое сочетание незаконченности и совершенства. Благодаря этому они, одновременно, убедительны и побудительны. Хоар не боится писать о свежей идее, но очень тщательно готовит публикацию, с большим количеством вариантов, которые он рассылает ближайшим коллегам с просьбами о критике и замечаниях. Себя он в шутку называет специалистом по переписыванию. Его публичные выступления лишены внешней эффектности, но предоставляют слушателю нечастую возможность поспевать за ходом мысли докладчика благодаря четкой структуре изложения, неторопливости и безупречному оксфордскому выговору. Когда Хоар говорит о мотивах или источниках своих исследований, обращает на себя внимание выношенный характер его идей, опирающихся на глубоко пережитый личный опыт. Эта независимость мышления не мешает ему быть внимательным слушателем, ценящим чужой аргумент.

Научная деятельность Хоара заслужила общее признание. В 1978 г. он был удостоен звания Выдающегося члена Британского вычислительного общества. В 1980 г. ему была присуждена премия имени Тьюринга «за фундаментальный вклад в определение и конструирование языков программирования». В 1982 г. он был избран членом Королевского общества.

В 1976 г. Хоар посетил Советский Союз, где выступал с циклом лекций перед специалистами и студентами Москвы, Ленинграда, Киева и Новосибирска.

А. П. Ершов

Цитируемые работы Ч. А. Р. Хоара

1. Об одном способе осуществления синтеза предложения при машинном переводе на основе синтагматического анализа. — В кн.: Машинный перевод и прикладная лингвистика. № 6. М.: 1-й МГПИИЯ, 1961, с. 80—88.

Продолжение см. на стр. 60



ПРОГРАММИРОВАНИЕ КАК ИНЖЕНЕРНАЯ ПРОФЕССИЯ

Ч. А. Р. Хоар

В прошлом благосостояние общества в значительной мере зависело от мастерства и трудолюбия ремесленников — кузнецов и мельников, ткачей и прядильщиков, столяров и плотников, сапожников и портных. Ремесленник обладал особым мастерством, которого не имели его заказчики, и которое приобреталось долгими голодными годами обучения у мастера. Обучение основывалось на подражании, практическом опыте, на пробах и ошибках. Он мало знал о научных основаниях своих приемов — о геометрии или хотя бы о черчении, о математике или хотя бы об арифметике. Он не мог объяснить, как он работает и почему работает именно так; тем не менее работал эффективно, один или в небольшом коллективе, и ему обычно удавалось выполнять то, за что он брался, в установленный срок, при фиксированной стоимости и результатах, которые удовлетворяли заказчика.

Сегодняшний программист во многом походит на вчерашнего ремесленника. Он осваивает свое ремесло во время короткого, но хорошо оплачиваемого ученичества в программистском коллективе, работающем над некоторым проектом, и совершенствует мастерство практическим опытом, а не читая книги и журналы. Он мало знает о логических и математических основаниях своей работы. Он не любит объяснять и документировать свою деятельность. Тем не менее он работает эффективно, один или в небольшом коллективе, и иногда ему удается выполнять то, за что он берется, в установленный срок, при установленных затратах и удовлетворить заказчика.

В примитивном обществе далекого прошлого был и другой класс специалистов, от которых зависело благосостояние общества. Как и ремесленник, такой специалист был предан своему делу и его уважали — и даже боялись — удовлетворенные заказчики. Таких людей называли разными именами — шаман, знахарь, колдун, пророк, маг, заклинатель, жрец. Я буду называть его просто «жрец».

Жрец во многом отличается от ремесленника. Важнейшее отличие заключалось в том, что жрец был хранителем набора священных книг или свода свитков с заговорами, которые лишь он один был способен прочесть. Когда заказчик приходил к нему с новой проблемой, жрец обращался к своим книгам в поисках заговора или заклятия, которые эффективно работали в прошлом; заказчик должен был тщательно скопировать его и применять в строгом соответствии с предписанием. Малейшая ошибка при копировании или применении могла превратить заклятие в проклятие и принести много бед заказчику. Заказчик не надеялся понять природу ошибки или почему она прогневала духов — даже и сам жрец не обладал полным пониманием путей бога. Лучшее, что мог сделать заказчик, — это повторить заклинание с начала. Если оно не работало, приходилось возвращаться к жрецу за новым заклинанием.

Здесь мы сталкиваемся с другой чертой жречества. Если что-либо выходило не так, как требовалось, а это случалось довольно часто, виной тому было невежество заказчика, или его тупость, или его порочность, или его греховность. И никогда здесь не было вины жреца или его бога. Когда засуха сжигала урожай, жрец приносил в жертву вождя — никогда наоборот.

Нынешние программисты во многом походят на жрецов. Их называют по-разному — кодировщик, системный аналитик, кибернетик, алгоритмист, системщик, ведущий программист (я буду называть их просто «программистами»). Наши алтари скрыты от непосвященных, каждый в своем храме с прекрасными кондиционерами; им день и ночь прислуживают преданные служители; и простой пользователь думает о них со страхом и почтением, соответствующими своему бессилию и зависимости. Еще ярче проявляется сходство в отношении наших священных книг, руководств по нашим языкам и операционным системам, без которых просто не подойдешь к машине. Каких-нибудь 30 лет назад лампы и провода наших ЭВМ заполняли полки и стены огромных залов, в которые входил программист с маленьким руководством в кармане — основные инструкции были выписаны на одном листке бумаги. Ныне все наоборот: программист приходит в комнату, полки и стены которой забиты томами руководств по программам; а на случай, если понадобится что-то быстро посчитать, он несет в кармане компьютер.

Рождение инженерии

Развитие технологии в течение последних веков привело к появлению нового класса специалистов — профессиональных инженеров. Важнейшая черта инженера — в том, как он приобретает свой статус специалиста. Подобно ремесленнику, он долго ходит в учениках у мастера, и, подобно жрецу, подвергается короткой церемонии посвящения; но этому предшествует многолетнее теоретическое обучение в школах и университетах. Образование инженера охватывает широкий круг предметов, включая математические основания дифференциального исчисления, вывод и решение сложных математических уравнений, физические принципы, лежащие в основе теории сопротивления материалов, а также специальные знания его узкой области, и множество испытанных методов проектирования и конкретных практических приемов. На протяже-

нии всей своей профессиональной деятельности инженер продолжает учиться, чтобы совершенствовать свое мастерство и не отстать от технического прогресса; для этого он читает новые книги и научные журналы и посещает лекции. Многие инженеры даже тратят целый год для того, чтобы овладеть современным уровнем знаний или приобрести специальность в развивающейся отрасли новой технологии. Старый ремесленник мог бы упрекнуть инженера в том, что он знает гораздо больше, чем требуется для повседневной профессиональной деятельности, но коллеги и заказчики понимают, что именно широта и глубина общего образования обеспечивают правильность решений и постоянный рост компетенции и авторитета инженера. Даже если он воспользуется какой-либо частицей своих знаний лишь раз в жизни, учение окупается многократно.

Мы осмеливаемся утверждать, что программирование уже переросло уровень ремесла, справилось с искушением превратиться в жреческую деятельность, и может рассматриваться как полноправная инженерная профессия. Мы имеем определенное право на подобное утверждение. Наши профессиональные организации выработали некоторый кодекс профессиональной этики, структуру и состав экзамена по специальности. Мы выполняем свой долг перед обществом, работая в государственных комиссиях по вопросам социальных последствий развития вычислительной техники, по вопросам собственности и трудоустройства населения. Наша работа пользуется большим спросом, наши заказчики и работодатели предлагают нам высокие заработки, от которых мы едва ли откажемся.

Но этого мало для действительно профессионального статуса. Где корпус профессиональных знаний, общих для всех образованных программистов? Где справочные библиотеки классических трудов по универсальным методам и частным приемам и специальным алгоритмам, для конкретных приложений и требований? Где теоретические (математические или физические) принципы, лежащие в основе повседневной деятельности программиста?

До самого последнего времени эти вопросы оставались без ответа. Теперь, наконец, кое-что начинает появляться. Можно указать на программы по информатике, предложенные ACM [1] и IEEE [2], как на корпус обще-программистских знаний, хотя доля программистов, сдающих экзамены по этим программам, все еще очень мала. Книга Дональда Кнута «Искусство программирования для ЭВМ» [3] является превосходной энциклопедией известных методов, но до сих пор вышли только три первых тома. И много ли программистов пользуется ими?

Наконец, мы лишь недавно пришли к пониманию математических и логических оснований программирования. Теперь мы уже можем начать строить спецификации программ с той же точностью, с какой инженер проектирует мост или шоссе, и на этой основе строить программы, соответствие спецификациям которых будет строго доказано; в этом соответствии можно быть так же уверенным, как инженер уверен в том, что построенный им мост не упадет. Методы, разработанные в таких работах, как «Структурное программирование» [4], «Систематическое программирование» [5], «Принципы построения программ» [6], «Дисциплина программирования» [7] и «Архитектура параллельных программ» [8] позволяют преобразовать таинственное и ненадежное ремесло программиста и привести его в соответствие с высокими стандартами современной инженерной профессии.

В действительности природа и последствия открытия оснований программирования еще шире. Оно подобно открытию древними греками аксиоматической геометрии — оснований для землемерной и картографической деятельности, а затем для проектирования и возведения зданий и мостов. Оно подобно открытию законов Ньютона и дифференциального исчисления — оснований для астрономии, а также для таких мирских начинаний, как навигация и управление артиллерийским огнем. Оно подобно открытию сопромата — основания для надежного и экономичного строительства железобетонных зданий, мостов и нефтехранилищ.

Символ N обозначает число (целое положительное).
Символ 1 обозначает единицу.
Символ a+1 обозначает следующий после a, или a плюс 1.
Символ = обозначает равенство.

1. $1 \in N$

2. $a \in N \;.\supset.\; a = a$

3. $a, b \in N \;.\supset:\; a = b \;.=.\; b = a$

4. $a, b, c \in N \;.\supset:\; a = b \;.\;.\; b = c \;:\supset.\; a = c$

5. $a = b \;.\; b \in N \;:\supset.\; a \in N$

6. $a \in N \;.\supset.\; a + 1 \in N$

7. $a, b \in N \;.\supset:\; a = b \;.=.\; a + 1 = b + 1$

8. $a \in N \;.\supset.\; a + 1 \neq 1$

9. $k \in K \;\therefore\; 1 \in k \;\therefore\; x \in N \;.\; x \in k \;:\supset_x.\; x + 1 \in k \;::\supset.\; N \supset k$

Эти известные всем математикам законы представляют сегодня основания арифметики и теории чисел. Где соответствующие законы программирования?

Большие программные проекты

Мы надеемся, что в будущем разработка и жизненный цикл крупных программных проектов радикально изменятся. Работа над проектом будет начинаться с того, что главный программист, подобно архитектору, обсудит требования с заказчиком. На основании своих знаний и опыта программист будет способен помочь заказчику понять свои истинные потребности и не включать в проект дорогостоящих компонент, полезность которых сомнительна или явно отрицательна. Уважая профессиональный статус программиста, заказчик будет с благодарностью принимать такое руководство. Подобное взаимопонимание существенно для любого взаимодействия между профессионалом и его заказчиком или нанимателем.

Спецификации

К этому моменту главный программист набросает эскиз общей структуры спецификаций проектируемого продукта в соответствии с требованиями заказчика. Этот эскиз будет играть такую же роль, как предварительные эскизы здания, которые рисует архитектор. Постепенно, в процессе упорядоченных тесных обсуждений с заказчиком, будут выстраиваться детали проекта. В результате будет выработана полная и лишенная неоднозначностей спецификация всего конечного продукта, состоятельность которой можно бу-

Отрывок из американского стандарта языка PL/1. Лишь высочайшие из верховных жрецов могут расшифровать эти иероглифы.

Шаг 7.2. Пусть g — <поколение>, которое непосредственно входит в abuf. Если abuf содержит <ключ>, положить k равным этому <ключу>, а csvb — непосредственной компоненте: иначе положить k и csvb равным <absent>. Выполнить построение—записи (g, k) со значением kr.

Шаг 7.3. Если fi включает <keyed>, то если k совпадает с каким-либо <ключом> в <наборе — данных>, обозначенном <обозначением—набора—данных> в fi, или если k не имеет смысла в данной реализации, то выполнить особая—ситуация—ввода—вывода (<key—condition>, fv, csvb).

Шаг 7.4. Выполнить вставка—записи (kr, fv) со значением pos.

Шаг 7.5. Заменить непосредственную компоненту <текущей—позиции> в fi на pos.

Шаг 7.6. Выполнить очистка (g) и исключить abuf из fi.

Шаг 8. Положить dd равным <описателю—данных> непосредственно содержащемуся в <переменной> из <определения>, обозначенного cdp. Выполнить вычисление—описателя—данных—для—размещения (dd) со значением edd.

Шаг 9. Выполнить определить—размер (edd) со значением <целое> int. Если int неприемлемо для данной реализации, то выполнить особая—ситуация—ввода—вывода <record—condition>, fv, chs).

Шаг 10. Выполнить размещение (edd) со значением g.

Шаг 11. Положить desc равным <описателю—данных>, который просто содержит <указатель> без промежуточных узлов. Положить epsog равным <месту—результата>, которое содержит <поколение> в <определенном—указателе> в els. Положить agv равным <составное—значение>, которое содержит <указатель> g. Выполнить назначение (epsog, agv, desc).

Шаг 12. Положить α равным <определению>, обозначенному <обозначением—определения> в els.

Шаг 12.1. Если <составной—тип> g содержит <структурный—тип>, выполнить инициализацию—ссылок (g).

Шаг 12.2. Выполнить инициализацию-поколения (g, d).

Шаг 13. Положить abuf равным <выделенному—буферу> : <поколение> g. Если fi содержит <ключевой>, то присоединить kk к abuf. Присоединить abuf к <открытию—файла> в fi.

Шаг 14. Продолжить нормальную обработку.

8.6.4. Оператор замены (REWRITE).

Назначение: <оператор—замены> вызывает замещение существующей <записи> или <записи—с—ключом> в <наборе—данный>.

8.6.4.1. Выполнить—оператор—замены.

<обработанный—оператор—замены> : : = <файловое —значение> ((<ключ>) <обработанный—парамерт—from>

Действие: выполнить—оператор—замены (rws), где rws есть <оператор—замены>.

Шаг 1. Положить erws равным <обработанному—оператору—замены> без подчиненных узлов.

Шаг 2. Шаги 2.1—2.3 можно выполнять в произвольном порядке.

Шаг 2.1. Положить f равным непосредственной составляющей <параметра—файл > в rws. Выполнить вычисление—параметра—файла (g, d) со значением <файловое—значение> fv. Присоединить fv к erws.

Шаг 2.2. Если rws содержит <параметр—файла> то выполнить вычисление—параметра—from (fr) со значением <обработанный—параметр—from> efo и присоединить его к erws.

дет формально доказать. Эта спецификация будет играть в дальнейшем ту же роль, что схемы в инженерном проектировании или планы и сечения в архитектуре.

Прежде чем дать согласие на реализацию, заказчик, без сомнения, захочет просмотреть и проверить полную спецификацию. Боюсь, что здесь он будет несколько шокирован. Он увидит не красивые картинки и рисунки, а множество математических определений и формул и логических доказательств, которые вряд ли будут понятны ему. Одна из главных проблем программирования заключается в том, что наши технические и структурные решения практически невидимы; ничто из того, что можно увидеть в законченной программе, нельзя заранее наглядно проиллюстрировать. Этот печальный факт объясняет как живучесть, так и полную бесполезность блок-схем программ.

Рассмотрение аналогий с другими профессиями может помочь найти способ уничтожить этот коммуникационный барьер между программистом и заказчиком. Прежде чем начнется строительство здания, архитектор берет спецификации и делает по ним ряд перспективных рисунков или даже объемных моделей, чтобы заказчик мог их посмотреть и сравнить со своими потребностями. Прежде чем начнется массовое производство изделия, инженер изготовляет серию работающих образцов, которые можно подвергнуть строгим всесторонним испытаниям в разнообразных условиях. В будущем главный программист, сам или с помощью своей команды, сможет сделать подобную модель разрабатываемого продукта.

Во-первых, на основании формальной спецификации будет написан ясный, полный и состоятельный комплект пользовательской документации, которая точно укажет, как пользоваться программой, и как программа будет вести себя во всех возможных ситуациях, включая ошибочные. Эти руководства обязательно будут проиллюстрированы подробными примерами типичных случаев, и эти примеры будут основаны на хорошо организованном и проиндексированном описании всех средств, предоставляемых программой. Эти описания разъяснят, почему и зачем нужны те или иные средства, как правильно применять их вместе с другими средствами, и что делать в случае ошибки. Эти руководства дадут потребителю полное представление о том, как будет выглядеть его программа и что она сможет для него сделать, еще до того, как программисты напишут первый оператор. Основанные на простой математической модели, руководства будут гораздо яснее, полнее и короче, чем описания, которые мы пишем сегодня — так же, как законы Ньютона короче и яснее, чем таблицы движения планет Тихо Браге.

Одновременно с составлением описания главный программист или кто-либо из его коллег может построить прототип разрабатываемой программы — всей или наиболее важных ее частей. Этот прототип можно быстро запрограммировать на интерпретаторе для небольшой модельной базы данных, которая полностью размещается в ОЗУ: модель можно реализовать на более мощной ЭВМ, чем та, для которой разрабатывается система. Заказчик сможет поэкспериментировать с такой «уменьшенной моделью» проектируемого продукта, проверить детали проектных решений и предложить изменения до того, как проект вступит в стадию реального проектирования и реализации.

Построение моделей и прототипов потребует дополнительных затрат. Однако предоставляемая заказчику возможность внести изменения в проект окупает эти затраты, особенно при разработке больших и важных систем. Исправление ошибок обходится гораздо дороже, когда они уже встроены в бетон мегабайтов программ.

На этом этапе необходимо выполнить еще одну важную работу. На основании исходных требований заказчика и формальной спецификации проекта разрабатывается и включается в контракт последовательность исчерпывающих тестов для приемочных испытаний. Часть этих тестов будет держаться в секрете, чтобы у разработчиков не было искушения ориентироваться на выполнение системой определенных тестовых примеров, а не на ее соответствие спецификациям. Необходимая строгость этих засекреченных тестов может быть обеспечена только соответствующей математической строгостью формальных спецификаций. В этом случае, если разработанная система не сможет выполнить тест, а разработчики будут оспаривать его адекватность, любой квалифицированный логик или математик сможет решить, кто прав.

Реализация

На следующем этапе начинается работа над детальным проектом программы, соответствующей согласованной и отлаженной спецификации. Выделяются основные компоненты проекта и с математической точностью определяются интерфейсы между ними. Вероятно, часть необходимых компонент можно будет взять или адаптировать из библиотеки модулей, описанной в учебниках по программированию. Остальные компоненты специфицируются так же тщательно и тем же способом, как ранее составлялась спецификация всей системы. Однако самое важное при разработке спецификаций заключается в том, что главный программист должен с помощью математического доказательства убедить себя и своих коллег в том, что если каждая компонента будет удовлетворять своим спецификациям, то система, полученная в результате сборки компонент, будет удовлетворять спецификации, утвержденной заказчиком. В будущем это будет считаться само собой разумеющимся, точно так, как сегодня никто не сомневается в том, что части моста, независимо изготовленные по заданным размерам, подойдут друг к другу при сборке на месте. Так мы надеемся уничтожить характерный для многих сегодняшних проектов этап так называемой «интеграции системы», на котором идет мучительный поиск и исправление ошибок в интерфейсах компонент системы. Этот этап — самая дорогая и неуправляемая фаза крупного проекта; к тому же она наступает в конце разработки, когда начинают поджимать договорные сроки.

Почему так дорого обходится отладка, особенно на этапе интеграции системы и позже, во время ее эксплуатации? Причина ясна: ошибки, проявляющиеся на этом этапе, столь тонкие, что они оказались не замечены проектировщиками, когда проект был еще прост и можно было выбирать. Они оказались не замечены и программистом, который вкладывал все свои интеллектуальные силы в

ЭВМ — чрезвычайно гибкий и мощный инструмент; многие понимают, что их применение уже меняет лицо Земли... Но возможно, что в качестве инструментов ЭВМ лишь поверхностно повлияют на нашу культуру; я уверен, что более глубокие изменения повлекут их интеллектуальные возможности. (Возможности в сфере интеллектуальной деятельности)

— Э. В. Дейкстра, 1972

каждую отдельную строчку текста программы. Теперь их надо найти в контексте миллиона строк программы и исправить; при этом требуется изменить как можно меньше строк из этих миллионов! Не удивительно, что сопровождение программы часто обходится во много раз дороже, чем первоначальная разработка. Мы надеемся, что применение новых методов спецификации и проектирования, основанных на математике и логике, позволит избежать этих затрат, с самого начала не допуская ошибок.

Когда проделана достаточно большая работа над проектом, можно приступать к организации программистских групп и составлять планы и графики для того, чтобы оценить объем и эффективность конечного продукта и, прежде всего, сравнить ожидаемые затраты с предварительными оценками. Это соответствует анализу материальных затрат в строительстве и требует как опыта и интуиции, так и математической квалификации. Как бы то ни было, эти оценки окажутся точнее и надежнее, чем бывает сегодня, так как они будут основаны на полной, состоятельной и неизменной спецификации проекта.

Наконец, проект готов для воплощения. Теперь можно привлечь большие коллективы программистов, возможно из различных фирм или организаций; все они могут работать одновременно над различными частями проекта без дополнительных консультаций и обсуждений. Каждый программист будет применять стандартные методы пошаговой разработки для того, чтобы его программа соответствовала спецификации, а риск ошибки был бы минимален. Когда он составит доказательство правильности программы (и доказательство, и программа проверяются и подписываются высокооплачиваемым контролером), только тогда текст программы вводится в машину.

Сдача в эксплуатацию

Когда текст программы написан, оттранслирован и программа загружена в ЭВМ, она подвергается тестам разработчика, которые она, как правило, пройдет сразу же. Затем она предъявляется заказчику и проходит его секретные тесты. Так как документация уже давно была доступна для изучения, можно сразу же приступать к эксплуатации системы. И начинается бессбойная эксплуатация системы. Как — никогда никакой ошибки? Ну, ладно, скажем, почти никогда! В тех редких случаях, когда ошибка все же произойдет, специальное всестороннее расследование определит причину ошибки и человека, ответственного за нее. Будет сделано заключение о том, является ли произошедший сбой разовым явлением или он связан с некоторой принципиальной ошибкой в логике проекта или в методах реализации. В последнем случае комиссии экспертов придется заново проверить значительную часть документации, доказательств и текстов программ, прежде чем система будет повторно сдана в эксплуатацию. Выплата неустоек заказчику будет способствовать тому, что такая ситуация будет редкостью.

Весьма вероятно, что в течение эксплуатации системы потребности пользователя будут изменяться, и система должна меняться в соответствии с ними. Так как программа будет понятно структурирована и полностью документирована, будет несложно определить, какие части проекта и текста программ необходимо модифицировать в соответствии с изменением требований заказчика. Так как для каждого модуля будут явно выписаны условия его применения и результаты его выполнения, будет относительно просто доказать, что если новая часть программы выполняет эти соглашения, ее включение не нарушит надежности системы в целом. Если изменения необходимо влекут нарушение некоторых условий, необходимых для работы существующих компонент системы, будет легко определить все места программы, зависящие от этих условий, и изменить их соответствующим образом. Если же предложенные модификации противоречат фундаментальной структуре программы, программист проанализирует возможность внесения глобальных изменений; в случае, если это окажется возможным, можно будет заранее определить объем и место необходимых исправлений и сразу же оценить, во что обойдется эта работа. Таким образом, удастся избежать неожиданностей, к которым приводит хаотическое внесение исправлений и расширений при принятом сегодня стиле сопровождения программного продукта.

На этом я закончу описание жизненного цикла программы в будущем. В этом описании я не упомянул один важнейший элемент современного программирования — ошибки в программе (bugs). Я не говорил о них потому, что их просто не будет. Исчезнет само понятие ошибки в программе. Будут искоренены условия их зарождения и процветания. Каждый шаг составления спецификаций, проектирования и написания текста программы будет проверяться с математической строгостью.

В любой сфере деятельности профессионала можно узнать по его умению организовать свою работу и применяемым методам, чтобы исключить случайные ошибки. Летчики, как правило, не разбивают самолеты. Хирурги, как правило, не убивают своих пациентов. Мосты, которые строят инженеры, как правило, не падают в реку. Пока все программисты не приобретут такой же уровень точности и ответственности, наше право на статус профессионалов будет вызывать сомнения. Каждый раз, когда кто-то обвиняет «машину» в ошибке, сделанной программистом, он порочит нашу профессию. Каждый раз, когда поставщик программного обеспечения официально снимает с себя ответственность за ущерб, наносимый непосредственно ошибками в его товаре, он порочит нашу профессию. Мы должны всегда признавать, что ответственность за ошибки несет программист, а не точная, но бессмысленная машина. Мы должны требовать юридического оформления ответственности за вред, причиняемый ошибками в программном продукте.

Мои замечания относятся, конечно, только к разработке больших и важных систем. В разработках меньшего масштаба многие этапы могут быть совмещены или опущены. Что же касается совсем маленьких программ (например, таких, которые пишутся на один раз), к ним то, о чем я говорил, вообще не относится. При построении песочного замка на пляже незачем использовать сопромат. Но никто не пригласит победителя конкурса песочных замков на пост главного архитектора комплекса высотных зданий.

Имеет смысл надеяться, что в следующем столетии союз программирования и математической логики будет столь же плодотворен, как союз математического анализа и физики в прошлом. Создание этого союза требует понимания обеих дисциплин и чувства математической красоты.

— Джон Мак-Карти, 1967

Сравнение с другими инженерными дисциплинами

Мое описание планирования работ по разработке крупного программного изделия во многом повторяет то, что принято в более традиционных инженерных областях. Лю-

бой инженерный проект проходит различные стадии — техническое задание, эскизное проектирование, экономическое обоснование, техническое проектирование, изготовление рабочих чертежей, изготовление прототипа, испытания, ОТК и прочее. Проходит много лет, прежде чем проект доходит до цеха.

Многие современные вычислительные центры организованы в соответствии с подобным разделением труда между системными аналитиками, алгоритмистами, составителями документации, программистами, и, наконец, группой сопровождения. Однако очень часто это, на первый взгляд логичное, разделение приводит к весьма неприятным проблемам. Группа сопровождения постепенно растет и перерастает по числу занятых все остальные подразделения, вместе взятые. При этом найти программистов для этого скучного, малопрестижного и часто низкооплачиваемого занятия становится все труднее. Причина этого, возможно, заключается в том, что, в отличие от того, как это принято в инженерной практике, порядок взаимодействия между перечисленными группами недостаточно строго определен, и практически отсутствует система последовательного контроля проектной документации при ее переходе с уровня на уровень. Каждая группа старается как может, а в конце цепочки стоят программисты группы сопровождения; они-то и собирают осколки.

Мне кажется, что логическая точность и полнота проектной документации, передаваемой одной группой разработчиков другой, должны удовлетворять требованиям, принятым в математике. Группа, получающая документацию, должна обладать интеллектуальными средствами для проверки ее состоятельности; у нее должно быть право — или даже обязанность — не принимать документацию, не соответствующую принятым требованиям. Разбор конфликтных ситуаций должен входить в обязанности технического руководителя проекта или проектной организации, обладающего опытом и компетенцией, необходимыми для профессионального решения возникшей проблемы. То, что руководители вычислительных центров и отделов обработки информации часто назначаются, исходя из их высоких достижений в электронике или экономике, приводит к печальным последствиям. Такие руководители имеют недостаточное представление о природе программирования и еще меньше о логических и математических методах, необходимых для руководства и управления им. Введение новых приемов принесло бы пользу в первую очередь таким руководителям; возможно, что именно поэтому, именно они зачастую являются наиболее упорными противниками перемен.

Надежность

Казалось бы, нам должно быть легче, чем другим инженерам, обеспечить высокий уровень качества и точности, так как сырье, с которым мы работаем, гораздо проще, имеется в достаточном количестве и весьма надежно. Ведь материал, с которым мы работаем, это биты ОЗУ и регистров, дисков и лент наших ЭВМ. Наша проблема не в том, что материала не хватает, а в том, что его слишком много. Эти биты обрабатываются в точном соответствии с нашими командами со скоростью миллионы операций в секунду, и в течение недель и месяцев не возникает ошибок; за сбои оборудования отвечаем не мы, а инженеры.

В успехе применения ЭВМ в научных исследованиях или обучении должны сыграть существенную роль результаты, методы и сам дух формализма математической логики.
— Хао Ванг (Hao Wang), **1967.**

Именно поэтому программирование должно стать самой надежной областью техники. Нам не приходится заботиться о неисправных деталях, неумелых рабочих, о бурях, землетрясениях или других стихийных бедствиях; нас не заботит износ деталей и старение металла. Все наши проблемы являются результатом нашего излишнего самомнения и нашей безответственности, нашей неспособности распознать математические и теоретические основания программирования, и нашей неспособности основывать на них нашу профессиональную деятельность.

Правда, инженеры имеют некоторое преимущество по сравнению с нами. Ведь величины, с которыми они работают — расстояние, температура, напряжение, меняются непрерывно. Поэтому они могут удостовериться в надежности своего изделия, проверив его в экстремальных условиях. Мы делаем то же при отладке программ, но в нашем случае толку от этого немного. Во-первых, нам приходится иметь дело с невероятно большим числом переменных; во-вторых, эти переменные принимают дискретные значения, для которых неприменимы традиционные методы интерполяции и экстраполяции. Если программа правильно работает для ноля и для 65535, нельзя уверенно утверждать, что она будет так же правильно работать для промежуточных значений, если это не подкреплено логическим рассуждением, основанным на тексте самой программы. Если же такое доказательство имеется, то и тесты не нужны. Именно поэтому, в качестве необходимой предпосылки улучшения нашей профессиональной деятельности, мы должны научиться формально анализировать наши программы, уметь доказывать их правильность еще до того, как они вводятся в ЭВМ, чтобы быть уверенными в том, что они не только пройдут отладочные тесты, но будут правильно работать всегда.

Структура

У инженеров других областей есть еще одно преимущество. Разделяя сложный проект на несколько независимо проектируемых компонент, он может использовать пространственное взаиморасположение частей, чтобы исключить неприятные неожиданности в результате их случайного взаимодействия. Если части вообще не имеют физических связей, это можно определить простым визуальным анализом. Так, поворачивая автомобиль влево, мы можем быть уверены, что это не будет иметь непосредственного влияния на зажигалку, зеркало заднего вида или карбюратор. Если наблюдается взаимодействие такого рода, все согласятся, что это дело сложное, и понять и исправить его будет непросто.

В программировании же у нас нет аналога пространственному разделению. В двоичной машине любая команда может изменить содержимое любой ячейки памяти машины, в том числе содержащей команды. И если это изменение будет сделано по ошибке при выполнении хоть одной из миллиарда команд, последствия для работы всей программы абсолютно непредсказуемы. Бессмысленно надеяться, что предварительный визуальный анализ содержимого памяти поможет нам убедиться в том, что такого взаимодействия не произойдет. В двоичной программе нет ни структуры, ни изолированных частей, кроме тех, которые внесены и старательно поддерживаются дисциплинированным программистом.

Несмотря на это от программиста часто требуют внесения изменений в работающую программу. Единственный быстрый способ сделать это — вставить в середину программы несколько команд, и тем самым разрушить тщательно установленные границы компонент и нарушить все структурные предположения, на которых основывалась изначальная разработка. Любому инженеру покажется бредом предложение связать рулевой механизм с карбюратором или лентопротяжку с плавающим процессором. Программист же на все согласен: в результате непредсказуемые побочные эффекты порочат имя нашей профессии.

Частичное решение проблемы лежит в использовании языков высокого уровня, таких, как Алгол-60, которые имеют надежные средства контроля за контекстом, областью действия и типом переменных. В таких языках программист определяет структуру программы, указывая, какие переменные могут использоваться или изменяться данным фрагментом программы. Транслятор автоматически проверяет соблюдение описанной дисциплины во всей программе; это дает программисту примерно те же гарантии независимости частей программы, что анализ пространственного расположения компонент инженеру. Еще больше гарантий обеспечивает выполнение программ на таких машинах, как Burroughs 5500, на которых аналогичные проверки выполняются во время работы программы. В традиционных инженерных областях соблюдение таких элементарных положений техники безопасности требуется законом. Закон требует принятия мер, чтобы избежать нежелательного взаимодействия станка с телом оператора.

Инструменты

Еще одна беда программистов — низкое качество инструментов, которыми они работают. Я имею в виду языки программирования, операционные системы, служебные программы и стандартные библиотеки. Все эти средства в изобилии поставляются вместе с ЭВМ. Многие из них столь сложны, что программист только и делает, что изучает инструмент, а для работы на заказчика у него не остается ни сил, ни времени. Многие операционные системы так плохо продуманы, что в течение десяти лет появляется до 20 вариантов («версий») системы, прежде чем удается исправить все ошибки, заложенные в первоначальном проекте. Когда же лихорадка версий кончается, бедному программисту не удается спокойно вздохнуть и начать нормальную работу — ему предлагают (или требуют) перейти на только что появившуюся новую систему. В период становления новой инженерной области приходится мириться с неудобством, ненадежностью и непостоянством инструментов; однако постепенно формируется набор для повседневной работы. Перед программированием пока еще стоит задача создать надежные, постоянные, удобные и, главное, простые для понимания, изучения и применения средства программирования.

Хорошей мерой простоты инструмента является объем текста для его исчерпывающего описания. В настоящее время документация по программному обеспечению объемиста и непонятна. Я уверен, что многие наши проблемы будут разрешены, когда мы научимся писать такие программы, для которых будет легко составлять короткую и понятную документацию. Если инженер-электронщик найдет способ заменить схему из 30 элементов схемой из 20 элементов, значение его изобретения сразу же будет оценено и вознаграждено славой или деньгами. Если инженер-программист найдет способ заменить программу, для описания которой необходимо 100 страниц, программой, описанной на 20 страницах, его заслуга еще больше — ведь он экономит самый дорогой ресурс: не кремний, не золото, а наш бесценный человеческий интеллект.

Направления движения

Мое описание профессионализма программистов будущего может показаться лишь мечтой теоретика — приятной для наших заказчиков и несколько страшноватой для нас самих. Как мы можем добиться такого фантастического улучшения методов работы? Мы похожи на средневековых цирюльников, которые гордились остротой своих бритв и скоростью, с которой они могли выполнить любую работу — будь то бритье бороды, или удаление язвы... Представьте себе прием, который бы оказали они теоретику, спустившемуся из башни из слоновой кости, чтобы сказать им, что практическая хирургия должна основываться на долгом подробном изучении анатомии человека, на знании методов, разработанных великими врачами древности; что оперировать можно только в антисептических условиях, не имеющих ничего общего с грязной и пыльной цирюльней. Даже если бы они признали справедливость и необходимость подобных усовершенствований, как могли бы они их добиться? Как обучить всех парикмахеров основаниям хирургии? Двухнедельный курс Структурной Хирургии — максимум, что мы можем предложить сегодня. А требуется больше, гораздо больше.

Специальная литература

Прежде всего нам необходимы новые хорошие книги и журналы, по которым программисты и группы программистов могли бы познакомиться с понятиями математических доказательств, и которые бы демонстрировали их применение в повседневной работе над спецификациями, проектированием и написанием программ. (Такие книги уже начинают появляться [9—13].).

Предлагаемые в этих книгах методы иллюстрируются небольшими примерами. Многие авторы никогда не применяли пропагандируемые ими методы к большим программам. Это не недостаток, а необходимая черта исследовательской работы. Любое научно-техническое изобретение сначала испытывается на моделях, на полигоне или в аэродинамической трубе. Без моделирования исследования были бы недопустимо дорогими, а прогресс соответственно медленным.

Теперь же настало время попытаться применить формальные математические методы к крупномасштабным проектам. Лучше всего это может быть сделано в рамках сотрудничества университета или академической организации с промышленной компанией или фирмой, разрабатывающей программное обеспечение. Таким проектом может быть создание совершенно новой системы или реорганизация (или модернизация) существующего продукта, который утратил первоначальную структуру в результате исправлений и улучшений в процессе сопровождения. В таком сотрудничестве ученые познакомятся с наболевшими проблемами разработки больших программ, а опытные программисты, работающие в промышленности, практически освоят теоретически разработанные методы. Это будет передача технологии в самом лучшем смысле — выгодная для обеих сторон.

Образование

Как я подчеркивал, главный фактор распространения профессионального подхода — образование, обеспечивающее как широкое и глубокое понимание теоретических принципов, так и их практическое применение; такое образование могут обеспечить наши университеты и институты. Профессора и преподаватели считают своим почетным долгом быть на уровне последних достижений в своей области и развивать и расширять читаемые ими курсы, чтобы передать свои знания студентам. Большинство студентов, начинающих слушать лекции по информатике, умеют программировать, научившись основным приемам в средней школе. Придя в университет, они готовы воспринять математические принципы, которые помогут им справиться со сложностью программ, которые им придется проектировать, и обеспечить надежность их реализации.

В ближайшие десятилетия, когда выпускники факультетов информатики и кибернетики начнут практическую деятельность, нас ждет очень сложный период. Практически никто из научных и административных руководителей не будет обладать достаточной квалификацией, чтобы понять новые методы; им будет казаться, что молодые специалисты городят ерунду. Это может создать большие сложности в становлении нашей профессии. Но избрав иной

Я всегда считал, что разделение практики и теории — искусственно и вредно. Многое в практике вычислительной техники, при разработке как программного, так и аппаратного обеспечения unsound and clumsy, **так как люди, занимающиеся этим, не имеют ясного представления о фундаментальных принципах, лежащих в основе их работы. Большинство теоретических и математических работ бесплодно, так как не имеет точек соприкосновения с реальным программированием и вычислительной техникой. Одной из главных задач Группы исследования программирования Оксфордского университета** (Programming Research Group) **стало создание такой атмосферы, в которой подобное разделение не сможет иметь места.**

— Кристофер Стрейчи (Christifer Strachey), **1974**

путь, мы потеряем слишком много. Одно из главных достоинств метода математической абстракции в том, что он позволяет главному программисту или руководителю осуществлять действительный контроль над работой отдельных групп, не погрязая в болоте технических подробностей, куда часто стараются затянуть его программисты.

Для того, чтобы справиться с грядущими потрясениями, те ведущие программисты, которые хотят сохранять свое положение, должны постараться в достаточной мере овладеть новыми теоретическими методами, чтобы оказаться способными эффективно выполнять функции главных программистов и научных руководителей проектов и технических директоров компаний.

Один из способов такой профессиональной переориентации — посещение специальных курсов повышения квалификации по новому важному предмету. Так, инженер-электронщик возвращается в университет, чтобы освоить проектирование СБИС; химик, работающий в промышленности, посещает специальный курс по химии полимеров или генной инженерии в каком-либо передовом университете. Я уверен, что уважающий себя программист должен следовать примеру представителей традиционных технических дисциплин. Поэтому в Оксфордском университете мы ввели новый специальный курс, посвященный, в первую очередь, улучшению методов программирования и внедрению их в широкую практику. Аналогичный курс читается в Wang Institute в США.

В 1828 году, выступая по случаю дарования Королевской Хартии Институту Гражданской Инженерии, Томас Тредголд определил гражданскую инженерию как «искусство направлять великие силы природы на пользу и благо человека». С тех пор расцвели многие отрасли инженерии. Они подчиняют энергию и занимаются ее хранением и преобразованием или обработкой и переработкой вещества. Программисты работают не с энергией, не с веществом, а с чем-то гораздо менее осязаемым. Мы подчиняем себе информацию и занимаемся ее хранением и преобразованием. Когда природа нашей деятельности будет достаточно широко осознана как в программистской среде, так и вне ее, тогда наша профессия получит заслуженное уважение и признание в качестве отрасли инженерии. И я уверен, что в нашей отрасли, как ни в какой другой, теоретические идеи строгости и элегантности максимально и зримо окупятся — снижая затраты, увеличивая производительность и направляя великие вычислительные силы кремниевого кристалла на пользу и благо человека.

ЛИТЕРАТУРА

1. ACM Recommended Curricula For Computer Science and Informatuin Processing Programs in Colleges and Universities. 1968—1981, N. Y.: ACM Press, 1981.
2. 1983 IEEE Computer Society model Program in Computer Science and Engineering. IEEE-CS Press. Silver Spring. Md. 1983.
3. Knuth D. E. The Art of Computer Programming, Uol. 1, 2, 3, Addison-Wesley, Reading, Ma., 1974, 1981, 1973. (Рус. пер.: Кнут Д. Е. Искусство программирования для ЭВМ. М.: Мир, 1976, 1977, 1978).
4. Dahl O. I., Dijkstra E. W., Hoare C. A. R. Structured Programming. N. Y.: Academic Press, 1972. (Рус. пер.: Дал О., Дейкстра Э., Хоар А. Структурное программирование. М.: Мир, 1975).
5. Wirth N. Systematic Programming, Prentice-Hall, Englewood Cliffs, N. J., 1973. (Рус. пер. Вирт Н. Систематическое программирование. Введение. М.: Мир, 1977).
6. Jackson M. A. Principles of Program Design. N. Y.: Academic Press, 1975.
7. Dijkstra E. W. Discipline of Program Design, Prentice-Hall, Englewood Cliffs, N. J., 1976. (Рус. пер. Дейкстра Э. Дисциплина программирования. М.: Мир, 1976).
8. Hansen P. B. The Architecture of Concurrent Programms, Prentice-Hall, Englewcod Cliffs, N. J., 1977.
9. Alagic S., Arbib M. M. The Design of Well-structured and Correct Programs. N. Y.: Springer-Uerlag, 1978. (Рус. пер.: Алагич С.. Арбиб М. Проектирование корректных структурных программ. М.: Радио и связь, 1984).
10. Gries D. ed. Programming Methodology. N. Y.: Springer-Uerlag, 1978.
11. Jones C. B. Software Development, A Rigorous Approach, Prentice-Hall, Englewood Cliffs, N. J., 1980.
12. Welsh J., McKeag R. M. Structured Systems Programming, Prentice-Hall, Englewood Cliffs, N. J., 1980.
13. Gries D. The Science of Computer Programming. N. Y.: Springer-Uerlag, 1981.

Статья поступила 17 сентября 1984 г.

Окончание. Начало см. на стр. 52

2. Quicksort. — Computer Journal, 1962, 5:1, p. 10—15.
3. (with M. Foley). Proof of a recursive program: Quicksort. — Computer Journal, 1971, 14:4, p. 391—395.
4. Report on the Elliott ALGOL translator. — Computer Journal, 1962, 5:2, p. 127—129.
5. The Emperor's old clothes. The 1980 Turing award lecture. — Comm. ACM, 1981, 24:2, p. 75—83.
6. (with N. Wirth). A contribution to the development of ALGOL. — Comm. ACM, 1966, 9:6, p. 413—432.
7. Record handling. — In: F. Genuys (Ed). Programming languages. London: Academic Press, 1968, p. 291—347.
8. An axiomatic approach to computer programming. — Comm. ACM, 1969, 12:10, p. 576—580, 583.
9. Monitoras an operating system structuring concept. — Comm. ACM, 1974, 17:10, p. 549—557.
10. Parallel programming: an axiomatic approach. — Comouter Languages, 1975, 1, p. 151—160.
11. A calculus of total correctness for communicating sequential processess. The Science of Computer Programming. 1981, 1:1, p. 49—72.
12. Communcating sequential processes. Prentice Hall, 1985 (в печати).

УДК 681.322.1

А. А. Попов, М. М. Хохлов, В. Л. Глухман

ДИАЛОГОВЫЕ ВЫЧИСЛИТЕЛЬНЫЕ КОМПЛЕКСЫ «ЭЛЕКТРОНИКА НЦ-80-20»

Диалоговые вычислительные комплексы (ДВК) «Электроника НЦ-80-20» предназначены для применения в качестве *инструментальных машин* для отработки программного обеспечения встраиваемых микроЭВМ и *терминальных устройств* в вычислительных системах и сетях, системах сбора и обработки данных с ограниченными ресурсами, информационно-справочных и информационно-поисковых системах.

По способу взаимодействия с пользователем ДВК можно отнести к диалоговым, по представлению ресурсов — к персональным, по ориентации на область применения — к профессиональным ЭВМ.

Архитектура ДВК

При разработке ДВК учитывались следующие основные принципы:

модульность построения;

расширяемость, наращиваемость;

совместимость снизу вверх на аппаратном и программном уровнях с серийно выпускаемыми микро- и мини-ЭВМ «Электроника 60, 100/25,79», СМ-3, СМ-4;

встраиваемость аппаратных средств в существующие базовые программные средства;

возможность использования серийного стандартного оборудования и программного обеспечения, разработанного для ЭВМ данного класса.

Две первые модели ДВК [1] «Электроника НЦ-80-20/1 (ДВК-1) и «Электроника НЦ-80-20/2» (ДВК-2) отличаются только составом оборудования (рис. 1, 2). Основу комплексов составляет одноплатная микроЭВМ «Электроника НЦ-80-01Д», функциональные возможности которой (система команд, ОЗУ, ПЗУ, интерфейсы) эквивалентны микроЭВМ «Электроника 60 М1», а быстродействие выше в 1,5 раза.

Эти модели разрабатывались как автономные, замкнутые системы, предназначенные для использования в качестве настольного автоматизированного рабочего места. «Зашитый» в ПЗУ одноплатной микроЭВМ интерпретатор языка Бейсик, аналогичный перфоленточному варианту языка микроЭВМ «Электроника 60», позволяет использовать ДВК-1 для решения инженерных задач, не

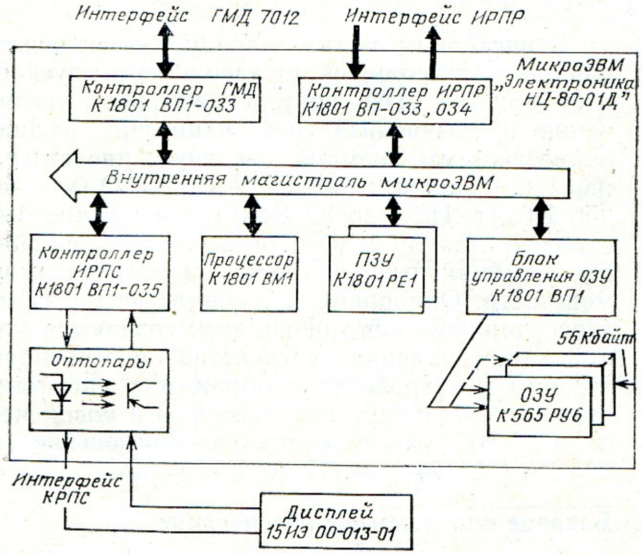


Рис. 1. Структурная схема ДВК-1

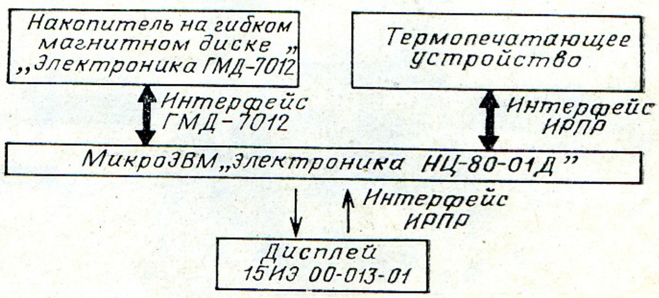


Рис. 2. Структура ДВК-2

требующих выдачи твердой копии. Вместо интерпретатора языка Бейсик в ПЗУ может быть «зашита» любая другая программа, реализующая конкретные функции пользователя.

Наличие в одноплатной микроЭВМ разъемов для вывода каналов, имеющих интерфейсы типа ИРПС, ИРПР, ИНГМД, а также разъемов, расположенных на задней панели блока символьного дисплея, позволяет подключать к ДВК-1 устройства, имеющие соответствующие интерфейсы, или использовать его в вычислительных системах или сетях в качестве терминальной микроЭВМ. Это относится и к ДВК-2 (см. рис. 2), в котором вместо термопечатающего устройства и НГМД «Электроника ГМД 7012» могут быть подключены любые другие устройства, имеющие соответствующие интерфейсы.

Меры по совершенствованию технологии разработки и производства печатных плат, увеличению их вычислительных мощностей при сохранении габаритных размеров, организация производства новых накопителей на гибких магнитных дисках позволили разработать новую модель ДВК — «Электроника НЦ-80-20/3» (ДВК-3, рис. 3—5), предназначенный для замены моделей ДВК-1 и ДВК-2. К особенностям новой модели относятся следующие:

уменьшение общих габаритных размеров, массы, потребляемой мощности;

меньшее число плат (вычислитель, контроллер дисплея символьного, контроллер накопителей на гибких магнитных дисках);

встроенный монитор и накопители на гибких магнитных дисках;

отсутствие внешних соединительных кабелей между устройствами.

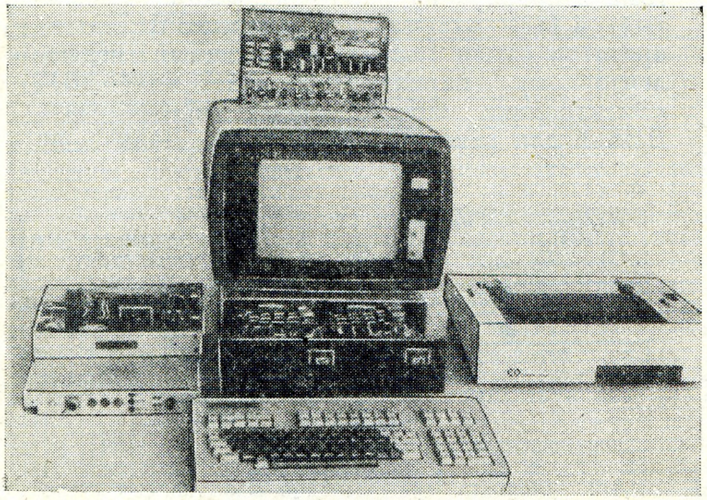


Рис. 3. «Электроника НЦ-80-20/3» (ДВК-3)

Основные характеристики ДВК-3, микроЭВМ «Электроника НЦ-80-01Д» и НГМД «Электроника 6021» приведены ниже.

«Электроника НЦ-80-20/3» (ДВК-3)

Быстродействие (Р-Р), операций/с	500 тыс.
Объем адресуемой памяти ОЗУ, Кбайт	64
Объем ПЗУ, Кбайт	8
Объем внешней памяти НГМД, Кбайт	440
Дисплей	алфавитно-цифровой
Число символов на экране	24×80=1920

Программное обеспечение — операционная система совместимая с ОС Фодос «Электроника 60», ОС Рафос мини-ЭВМ, СМ-3, СМ-4

Языки программирования	макро, Бейсик, Фортран, Паскаль
Интерфейсы:	ИРПР, ИРПС.

МикроЭВМ «Электроника НЦ-80-01Д»

Быстродействие (Р-Р), операций/с	500 тыс.
Объем адресуемой памяти ОЗУ, Кбайт	64
Объем пользовательского ПЗУ, Кбайт	8
Число уровней прерываний от внешних устройств	2

Система команд совместима с системой команд микроЭВМ «Электроника 60»

Накопитель на гибких магнитных дисках «Электроника 6021»

Число приводов	1
Число рабочих поверхностей	2
Число дорожек	40
Число секторов	11
Емкость сектора, Кбайт	220
Скорость вращения диска, об/мин	300
Время перемещения на соседнюю дорожку, мс	15
Время подвода головки, мс	50

В настоящее время, наряду с совершенствованием технологии производства выпускаемых моделей ДВК ведутся работы по увеличению их вычислительной мощности: расширение системы команд, введение диспетчера памяти, наращивание объема ОЗУ до 256 Кбайт, ПЗУ до 32 Кбайт, увеличение быстродействия до 1 млн. операций/с при меньших габаритных размерах и потребляемой мощности. Одновременно, совместно с заинтересованными организациями, создаются новые запоминающие устройства повышенной емкости и устройства отображения информации, которые будут использованы в новой модели ДВК, рассчитанной на применение в СЧПУ, САПР, АСУТП, АСНТИ и др.

Базовое программное обеспечение

Базовое программное обеспечение (БПО), ориентированное на различные области применения ДВК включает операционную систе-

му разделения функций и систему контроля работоспособности.

Операционная система разделения функций (ОС ДВК) является подмножеством системы РАФОС и аналогом системы ФОДОС. Предназначена для решения задач в реальном масштабе времени и разработки программ в режиме диалога. Дает возможность пользователю решать одновременно основную и фоновую задачи, организовывать обмен данными между периферийными устройствами и связь с ДВК, программировать, редактировать и отлаживать программы.

Операционная система включает управляющую систему, систему программирования, редактирования и отладки программ, файловую и сервисную системы. Ядром управляющей системы является ФВ-монитор, резидентная часть которого занимает 8 Кбайт оперативной памяти. Для организации выполнения задач ФВ-монитор требует от 24 до 56 Кбайт оперативной памяти. Управляющая система может работать и под управлением S_i-монитора, с резидентной частью объемом 2 Кбайта оперативной памяти, но только в однозначном режиме выполнения программ. Монитор выполняет функции копирования, стирания, переименования файлов, вывода каталога, запуска программы, задания даты, вызова прикладных, системных и сервисных программ и др. Все они задаются с помощью команд, набираемых на клавиатуре дисплея. Команды монитора имеют единый формат: «имя команды», «спецификация файла ввода», «спецификация файла вывода». В спецификациях файлов указывается: «имя устройства», «имя файла», «тип файла». Здесь «имя устройства» — одно из следующих устройств: накопитель на гибком магнитном диске, устройство печати, терминальное устройство и др.

Для файлов пользователя имя файла и его тип определяется пользователем, а для системных файлов имена задаются при разработке системы. Допускается спецификация файлов по умолчанию. В этом случае отдельные элементы спецификации могут отсутствовать. Разрешается также краткая форма записи команды, задание групповых операций, а также создание командных файлов для запуска одних и тех же последовательностей команд монитора.

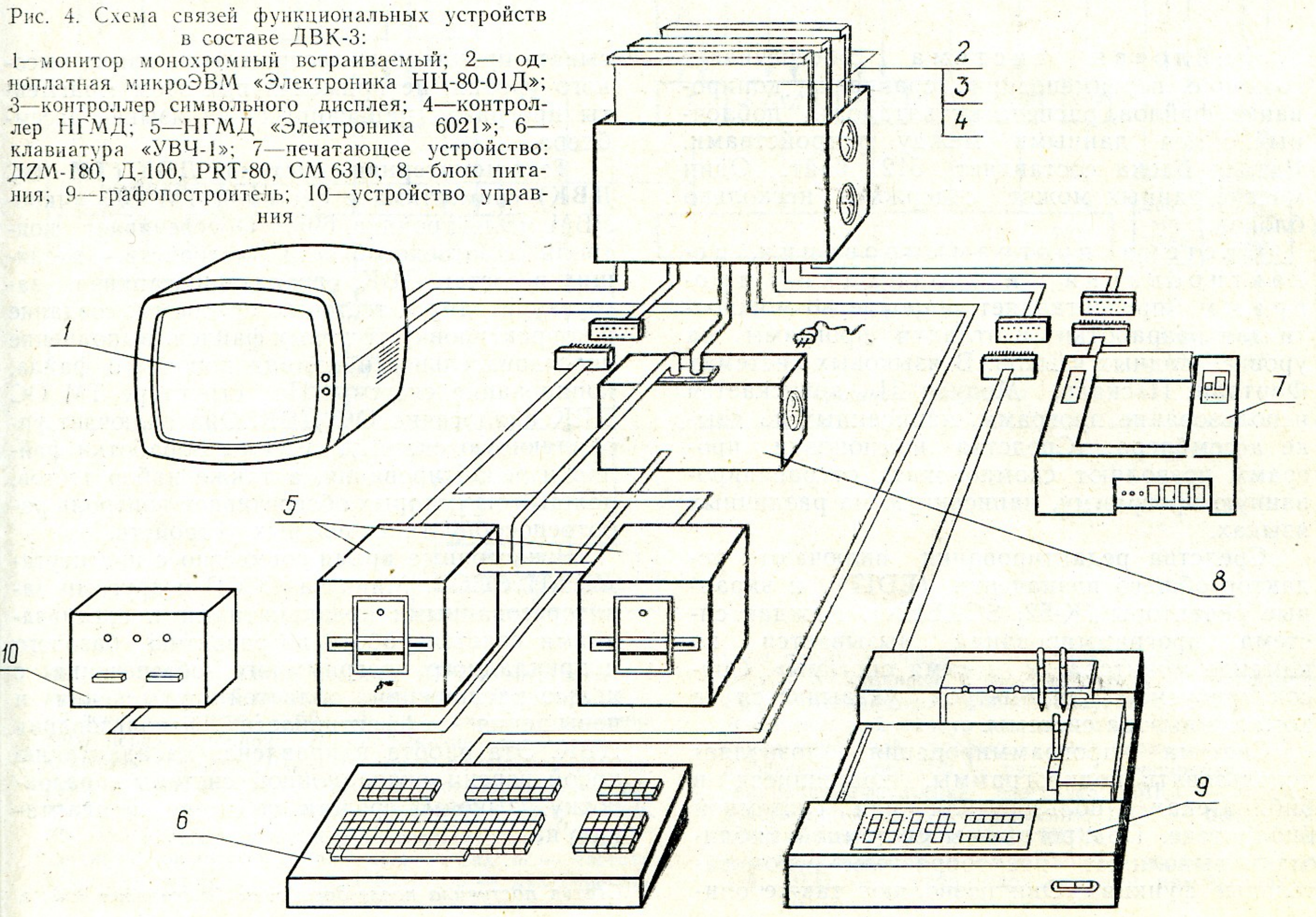


Рис. 4. Схема связей функциональных устройств в составе ДВК-3:
1—монитор монохромный встраиваемый; 2— одноплатная микроЭВМ «Электроника НЦ-80-01Д»; 3—контроллер символьного дисплея; 4—контроллер НГМД; 5—НГМД «Электроника 6021»; 6—клавиатура «УВЧ-1»; 7—печатающее устройство: DZM-180, Д-100, PRT-80, СМ 6310; 8—блок питания; 9—графопостроитель; 10—устройство управления

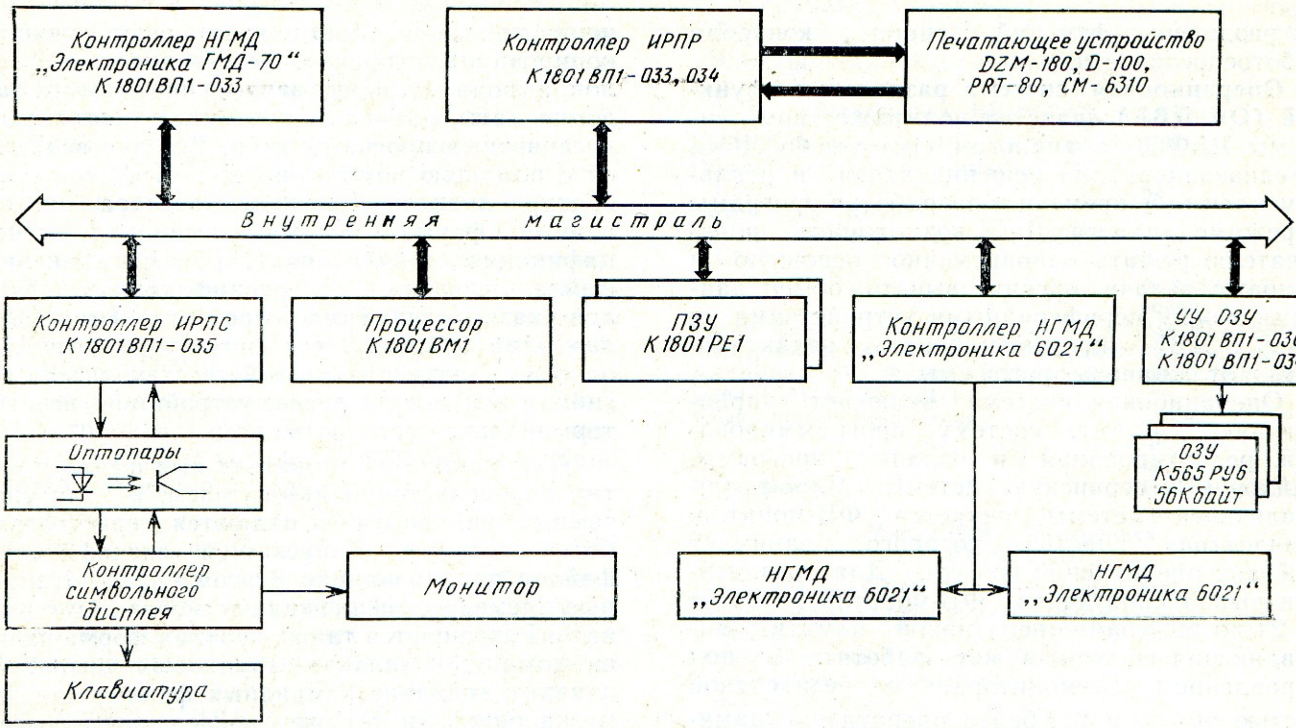


Рис. 5. Структурная схема ДВК-3

Файловая система обеспечивает создание, переименование, сравнение, копирование файлов, распечатку каталога, поблочный обмен данными между устройствами. Размер блока составляет 512 байт. Один массив данных может содержать несколько блоков.

Система программирования, редактирования и отладки программ предоставляет широкие возможности для разработки и отладки программ на уровне входных языков. В языковых системах Фортран, Паскаль, Модула II допускается использование программ, написанных на языке ассемблера. Средства компоновки программ позволяют скомпоновать оттранслированную программу, написанную на различных языках.

Средства редактирования включают редактор общего назначения (EDIT) и экранные редакторы (К-52, SCREEN). Каждая система программирования вызывается по команде монитора: R ␣ «имя системы». Список требуемых параметров указывается в команде вызова системы.

Система программирования позволяет использовать подпрограммы, хранящиеся в библиотеках пользователя или системной библиотеке. Подпрограммы системной библиотеки выполняют разнообразные вспомогательные функции. Они позволяют также описывать прикладные программы на языке высокого уровня, не применяя при этом фрагменты программ, написанных на языке ассемблера.

Тест-мониторная система ДВК (ТМ ОС ДВК) совместима с системой ТМ ОС микроЭВМ «Электроника 60». Обеспечивает контроль работоспособности устройств, входящих в состав ДВК, организацию архива, загрузку и запуск тестовых программ, создание и корректировку тестовых файлов, выполнение последовательности команд цепочного файла, копирование системы. По структуре ТМ ОС ДВК аналогична ОС ДВК. Она включает управляющую систему, системы обработки файлов и редактирования, а также набор тестов, каждый из которых обеспечивает контроль работоспособности отдельных устройств.

В настоящее время совместно с институтами АН СССР, Минвуза СССР и другими заинтересованными предприятиями и организациями ведется работа по развитию базового и прикладного программного обеспечения с целью расширения областей применения и повышения эффективности использования ДВК. Эта работа направлена на адаптацию новой версии операционной системы, разработку языковых средств, создание и адаптацию пакетов прикладных программ.

Статья поступила после доработки 10 сентября 1984 г.

УДК 681.32—181.4:621.31

Н. В. Колобродов, Л. С. Кусакина, В. С. Мельников, О. А. Федулова, Ю. В. Чугунников

МИКРОПРОЦЕССОРНОЕ УСТРОЙСТВО УПРАВЛЕНИЯ МОЩНОСТЬЮ КРУПНЫХ ПАРОВЫХ ТУРБИН

В ВЭИ им. В. И. Ленина разработано унифицированное всережимное *устройство* управления мощностью крупных паровых турбин (УУМПТ) на основе комплекса микросредств управляющей вычислительной техники (МСУВТ-В7). Структура *устройства* позволяет только за счет программного обеспечения изменять алгоритмы управления и функциональное назначение устройства. При проработке структуры УУМПТ были учтены все требования, предъявляемые к устройствам, входящим в систему автоматического регулирования частоты и мощности (АРЧМ) энергоблока, а также оказалось возможным и целесообразным реализовать ряд новых функций, выполнявшихся ранее специальными блоками: ограничения темпа задания [2] (ОТЗ); контроля за температурными напряжениями, возникающими при прогреве роторов турбин; определения допустимого по тепловому состоянию диапазона изменения нагрузки турбины [4]; диагностики состояния оборудования энергоблока.

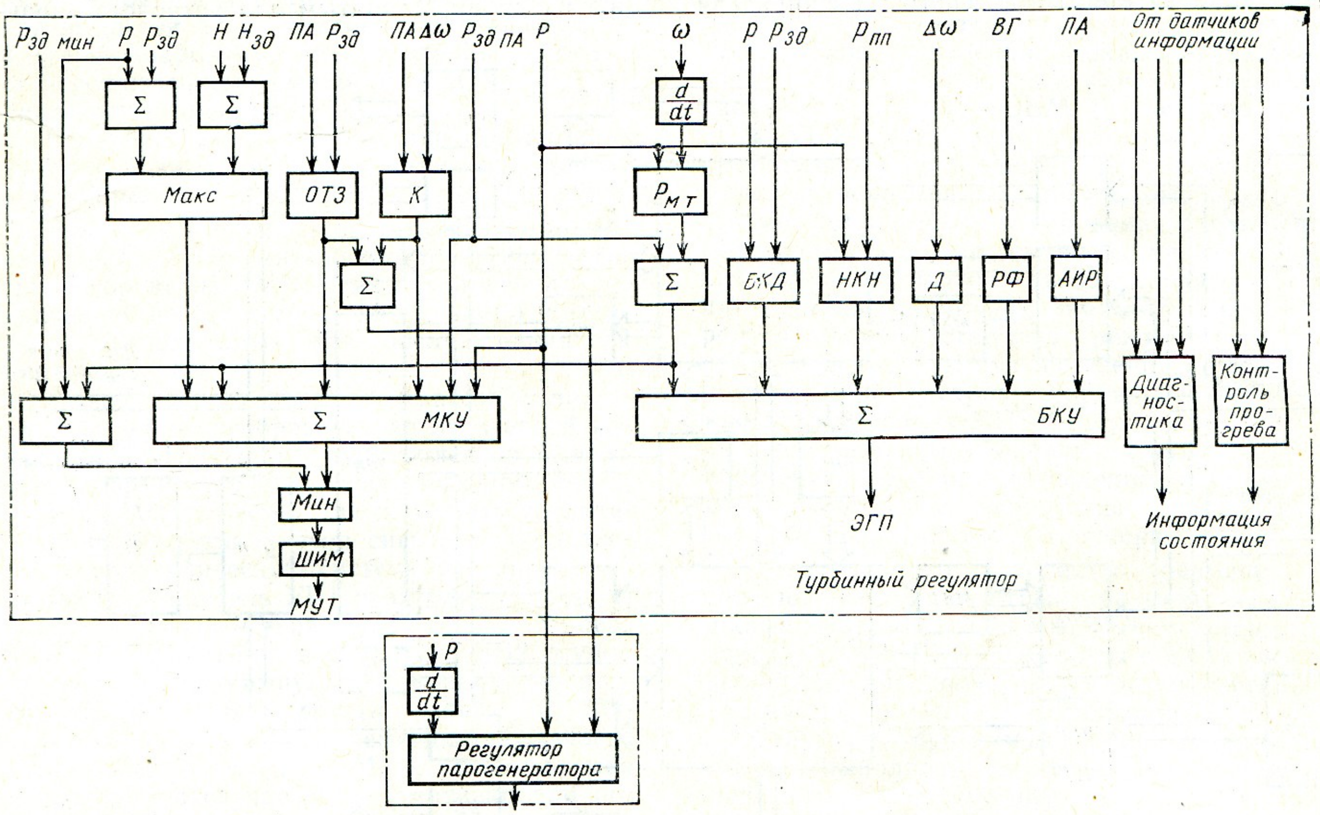


Рис. 1. Типовая схема системы АРЧМ энергоблока: $p_{з\ д\ мин}$ — заданное давление защитного регулятора; p — давление пара перед турбиной; $p_{зд}$ — заданное давление пара перед турбиной; H — положение регулирующих клапанов турбины; $H_{зд}$ — заданное положение регулирующих клапанов; ПА — сигналы противоаварийной автоматики; $P_{зд}$ — заданная мощность энергоблоку; $\Delta\omega$ — отклонение частоты в энергосистеме; $P_{зд\ ПА}$ — заданная мощность в аварийных режимах; ω — частота вращения ротора турбогенератора; $p_{пп}$ — давление пара в промперегревателе; ВГ — выключатель генератора; ОТЗ — ограничитель темпа задания; K — коэффициент; $P_{м\ т}$ — механическая мощность турбины; БКД — быстродействующий контур регулирования давления; НКН — канал начальной коррекции неравномерности; Д — дифференциатор; РФ — релейная форсировка; АИР — аварийная импульсная разгрузка; МКУ — медленнодействующий контур управления; БКУ — быстродействующий контур управления; ШИМ — широтно-импульсный модулятор; МУТ — механизм управления турбиной; ЭГП — электрогидравлический преобразователь

В типовой системе АРЧМ (рис. 1) предусмотрено формирование соответствующих управляющих воздействий на регулирующие органы турбины в режимах пуска и останова энергоблока, в нормальных [1, 2] и аварийных [5] режимах энергосистемы, а также взаимодействие с регулятором парогенератора и системой противоаварийной автоматики (ПА).

Устройство управления мощностью паровых турбин (рис. 2) должно функционировать с высокой надежностью. Для удовлетворения этого требования применен один из способов резервирования — дублирование микроЭВМ. Одна микроЭВМ является ведущей, а другая находится в горячем резерве. МикроЭВМ связаны между собой двунаправленной шиной данных, позволяющей контролировать и координировать вычислительный процесс.

В УУМПТ используется МСУВТ-В7 в шестиплатном варианте исполнения: одна плата процессора (ПМВ), две платы памяти (МВ), одна плата арифметического расширителя (ПСВ), две платы аналогового ввода-вывода (АВВ). Такой набор плат обусловлен большим объёмом обрабатываемой информации, представляемой как в аналоговом, так и в дискретном виде.

В каждой из микроЭВМ программными и аппаратными средствами формируются сигналы, несущие информацию о состоянии машины: превышение времени цикла обработки программ; обращение к несуществующей памяти; нарушение четности; отказ платы ПСВ; отказ платы МВ; отказ платы АВВ; отказ блока питания. Эти сигналы вводятся в блок контроля (БК), где по ним устанавливается режим работы ЭВМ1, ЭВМ2 и формируются сигналы о неисправности и аварии в УУМПТ. Кроме того, в БК, посредством светодиодной индикации, отражается состояние каждой микроЭВМ.

Для повышения помехозащищенности микроЭВМ и удовлетворения требованиям к уровню изоляции, принятым для устройств управ-

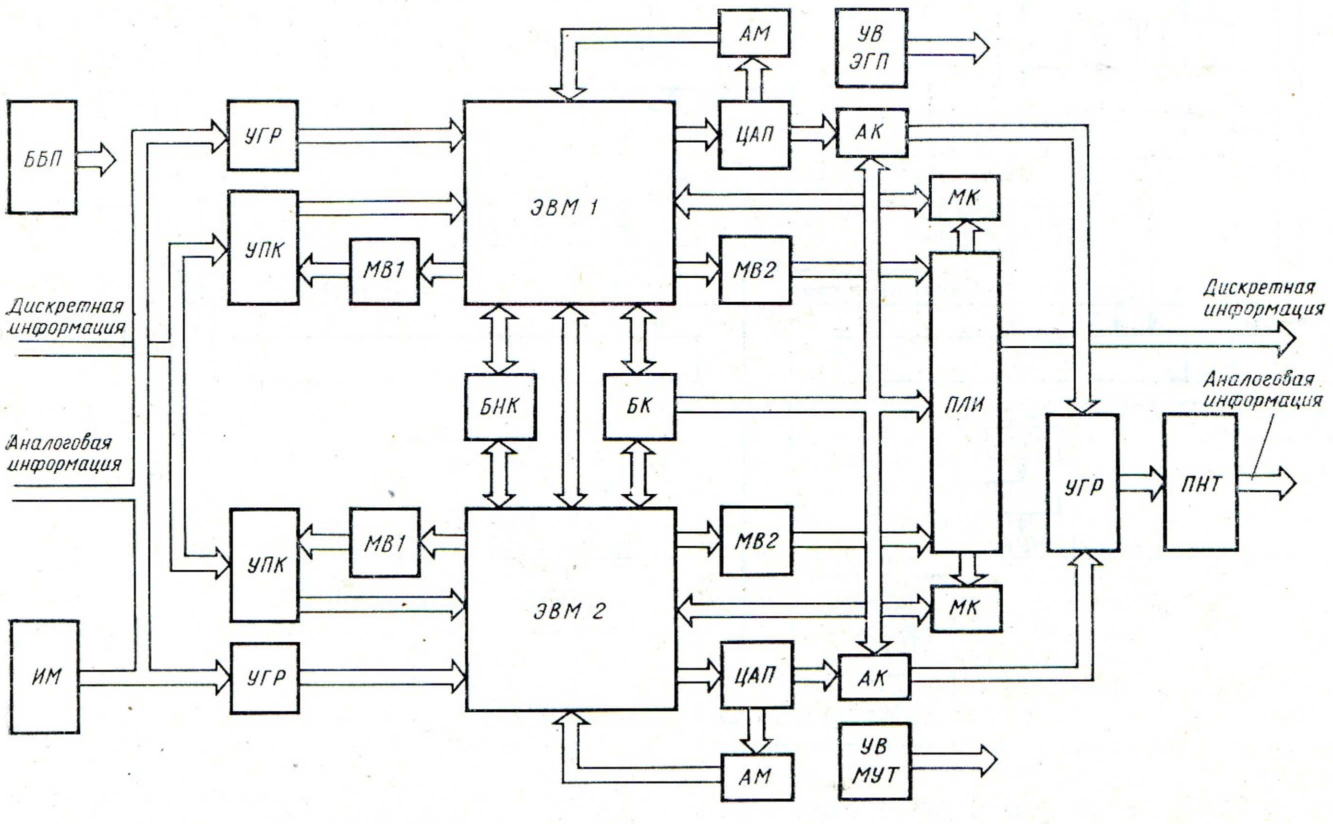


Рис. 2. Схема устройства управления мощностью паровых турбин:
ББП — блок бесперебойного питания; ИМ — измеритель мощности; УГР — устройство гальванической развязки; УПК — устройство приема контактов; МВ1 — мультиплексор ввода; БИК — блок набора коэффициентов; БК — блок контроля; АМ — аналоговый мультиплексор; ЦАП — цифро-аналоговый преобразователь; МВ2 — мультиплексор вывода; УВ ЭГП — усилитель выходного канала электрогидравлического преобразователя (ЭГП); АК — аналоговый коммутатор; УВ МУТ — усилитель выходного канала управления двигателем механизма управления турбиной (МУТ); МК — мультиплексор контроля; ПЛИ — преобразователи логической информации; ПИТ — преобразователи «напряжение — ток»

ления и регулирования электростанций, разработаны специальные устройства связи с объектом (УСО). УСО обеспечивают согласование и гальваническое разделение цепей ввода и вывода информации, представленной как унифицированными аналоговыми сигналами (±5 мА при нагрузке 2,5 кОм), так и дискретной информацией («сухой контакт»).

Для ввода в ЭВМ аналоговой информации используется устройство гальванической развязки (УГР), работающее по принципу «модуляции-демодуляции». Всего в ЭВМ вводится около 30 каналов аналоговой информации от датчиков режимных параметров и внешних устройств задания установок и 25 контрольных, дублирующих выходные аналоговые каналы УУМПТ. В каждую из ЭВМ аналоговая информация вводится через свое УГР.

Вывод аналоговой информации обеспечивают цифровые преобразователи (ЦАП), УГР и преобразователи «напряжение—ток» с унифицированными параметрами (ПНТ). Информация выдается по 24 аналоговым каналам от ведущей микроЭВМ. Блок контроля управляет переключением микроЭВМ и УСО через аналоговый коммутатор (АК).

Ввод дискретной информации осуществляется по 72 каналам через устройства приема контактных сигналов (УПК).

Для сокращения числа используемых линий параллельного интерфейса применен мультиплексор ввода (МВ1), обеспечивающий сокращение необходимого числа линий интерфейса, необходимых для обслуживания 72 каналов, до 12.

Вывод дискретной информации осуществляется через преобразователи логической информации (ПЛИ) и мультиплексор вывода МВ2. Формируется 48 дискретных команд.

В УУМПТ имеется два выходных усилителя, обеспечивающих ввод сигналов управления в гидравлическую систему регулирования турбины. Усилитель УВ1 реализует пропорциональные воздействия быстродействующего контура управления, а усилитель УВ2 осуществляет широтно-импульсное управление двигателем МУТ. В *устройство* входят измеритель мощности (ИМ) и схема дистанционного управления двигателем МУТ, используемая оператором в случае неавтоматизированного управления турбиной.

Для повышения надежности функционирования УСО введена система программно-аппаратного контроля. Входные УСО (УГР, УПК) контролируются программно путем сравнения одноименных сигналов, вводимых в разные ЭВМ.

Неисправность выходных аналоговых каналов выявляется при сравнении в ЭВМ кода, пропорционального сигналу, выдаваемому на ЦАП, и кода, соответствующего этому же сигналу и подаваемого на АЦП через дополнительное УГР, включенное на выходе соответствующего ПНТ. При этом контролируется весь тракт аналогового канала (ЦАП, АК, УГР, ПНТ) без выделения отказавшего устройства.

Для уточнения места отказа сигналы с выхода ЦАП через аналоговый мультиплексор (АМ) вновь подаются на вход АЦП ЭВМ. Это позволяет выявить отказ ЦАП и перейти на управление от второй ЭВМ. При неисправности в субблоках, следующих после ЦАП, формируется сопутствующий логический сигнал о выходе из строя данного канала. Выявление неисправности в каналах вывода дискретной информации также производится при сравнении в ЭВМ кода, выдаваемого на мультиплексор вывода, и кода, отражающего состояние контактов ПЛК и вводимого в ЭВМ через мультиплексор контрольных сигналов (МК).

Отказ УСО фиксируется в ЭВМ, анализируется и принимается решение об изменении режима или отключении. Каждый отказ отражается светодиодной индикацией и информацией на пульте микроЭВМ. С помощью пульта МСУВТ-В7 типа КЛОЗ и соответствующего программного обеспечения реализуется ряд функций, необходимых при наладке и эксплуатации устройства: высвечивание и изменение содержимого памяти; высвечивание и изменение состояния портов ввода-вывода; индикация отказов; индикация параметров процесса регулирования; установка режима регистрации в массиве памяти изменения во времени любого из параметров; управление режимом программирования ППЗУ и др.

Система бесперебойного питания (ББП) обеспечивает при отключении напряжения собственных нужд переход на питание от источника оперативного напряжения (=220 В).

Конструктивно устройство размещено в трех шкафах. Габаритные размеры каждого шкафа 800×800×2000 мм. УСО размещено в конструктивах БУК-6. Кассеты с УСО смонтированы на двух поворотных рамах, обеспечивая их двустороннее обслуживание.

Структура программного обеспечения позволяет выполнить все требования по быстродействию и точности, предъявляемые к системам АРЧМ. Функциональные программы, обеспечивающие расчет управляющих воздействий, выделены в специальный блок. Это дает возможность без особых затрат производить их корректировку и замену. В последнем случае может быть изменено и функциональное назначение устройства.

Подбор коэффициентов и констант, необходимых для расчета, обеспечивается с помощью кодирующих переключателей, разме-

щенных в блоке набора коэффициентов (БНК), или перепрограммирования кристаллов ППЗУ самой микроЭВМ.

Опытный образец устройства включен в эксплуатацию на блоке мощностью 300 МВт Литовской ГРЭС. Начато серийное изготовление аппаратуры управления мощностью крупных паровых турбин.

ЛИТЕРАТУРА

1. Давыдов Н. И., Маламед А. Д., Трахтемберг М. Д., Фотин Л. П. Система автоматического регулирования частоты и мощности блочных ТЭС с прямоточными котлами. — Теплоэнергетика, 1979, № 8, с. 2—6.

2. Горячева Ю. Л., Мельников В. С. Система управления мощностью турбоагрегата в нормальных и послеаварийных режимах энергосистемы. — В кн.: Автоматическое регулирование и управление в энергосистемах. — Труды ВЭИ, 1977, вып. 83, с. 93—99.

3. Жданова Л. В., Кусакина Л. С., Серебряков А. З. Принципы автоматического ограничения скорости изменения мощности энергоблока. — В кн.: Автоматическое регулирование и управление в энергосистемах. — Труды ВЭИ, 1983, с. 30—32.

4. Лейзерович А. Ш., Малев В. В. Аналоговое устройство для контроля за прогревом роторов паровых турбин. — Теплоэнергетика, 1980, № 7, с. 30—33.

5. Иофьев Б. И. Автоматическое аварийное управление мощностью энергосистем. — М.: Энергия, 1974. — 416 с.

Статья поступила 18 апреля 1984 г.

УДК 658.012

А. А. Кочетков, В. В. Крылов

ДИАЛОГОВАЯ СИСТЕМА АВТОМАТИЗАЦИИ МЕТЕОРОЛОГИЧЕСКИХ ИЗМЕРЕНИЙ

Успешное прогнозирование погоды, надежное осуществление авиаполетов во многом определяются непрерывным наблюдением за метеопараметрами нижнего слоя атмосферы от нулевой высоты до тропопаузы (∼ 10 км). В настоящее время изучение ведется с помощью шаров-зондов, оборудованных соответствующей аппаратурой. В ближайшем будущем для дистанционного зондирования нижней атмосферы предполагается широко использовать автоматизированные станции, которые дают возможность удешевить исследования, резко сократить интервалы между зондированиями. Работа одной из таких станций, получившей название ДИПАРМ (дистанционное измерение параметров атмосферы радиометрическим методом), основана на связи интенсивности переизлучения атмосферой в линиях поглощения с локальными метеорологическими характеристиками, такими как температура, давление, влажность. По измерениям интенсивности излучения атмосферы в различных спектральных участках и при необходимости под различными зенитными углами производится идентификация пространственного распределения метеопараметров. Более подробно физические и математические аспекты дистанционного метода зондирования, реализованного в системе ДИПАРМ, рассмотрены в работах [1—4]. Один цикл измерения профилей метеопараметров заключается в следующем (в скобках приведены интервалы длительностей этапов в зависимости от режима измерений):

проведение калибровки приемников излучения атмосферы эталонным источником (1...3 мин);

измерение излучения атмосферы на заданном множестве частот из интервала 20...100 ГГц и зенитных углов из диапазона 0...90° (1...10 мин);

проведение второй калибровки приемников (1...3 мин);

коррекция результатов измерений с учетом поправок калибровки;

расчет вертикальных распределений температуры, давления, влажности и параметров облачности по результатам измерений путем решения системы интегро-дифференциальных уравнений, описывающих связь между излучением и метеопараметрами (5...20 мин);

запись результатов измерений и расчета в архив и выдача протоколов в виде метеографиков и таблиц.

Необходимость двух процедур калибровки определяется особенностью приемников излучения и методики измерения. Приведенный цикл измерений повторяется через заданный интервал времени (10 мин и более).

Ручное вмешательство в процесс измерений сведено к минимуму, поэтому работать со станцией ДИПАРМ могут пользователи, не имеющие специальной подготовки. Оператор в процессе диалога с компьютером станции определяет необходимый режим измере-

ний, исходя из текущей метеорологической обстановки. Весь дальнейший процесс измерений происходит автоматически без вмешательства человека.

Аппаратные средства

Станция построена по блочно-модульному принципу с единой магистралью обмена данными при максимальном использовании серийных узлов и блоков (рис. 1).

Станция ДИПАРМ определяет распределение температуры, влажности и давления атмосферы до высот 10 км, на основе анализа собственного микроволнового излучения атмосферы.

В состав станции входят антенная система, многоканальный радиометрический приемник и система автоматизации обработки результатов измерений на базе микроЭВМ «Электроника 60» с внешней памятью на гибких магнитных дисках.

В качестве центрального процессора станции используется микроЭВМ «Электроника 60». Результаты измерений передаются в узловой вычислительный центр, оснащенный мини-ЭВМ СМ-4 для дальнейшей обработки и долговременного архивного хранения. Для задания сетки меток времени, необходимых для синхронизации всех процессов в станции, а также для привязки результатов измерений к астрономическому времени, используется таймер. Программное обеспечение и локальный архив измерений хранятся на гибких магнитных дисках накопителя «Электроника ГМД-70». Управление станцией ведется с терминала «Электроника 15-ИЭ-00-013», а протоколы работы и результаты измерений печатаются на устройстве печати DZM-180. Все перечисленные блоки изготавливаются серийно и входят в состав управляющего вычислительного комплекса 15-ВУМС-025-028.

Нестандартное оборудование станции включает многоканальный приемник с полосой 300 МГц, чувствительностью 0,5...1 К, настроенный на частоты 22,22; 37,5; 53,5; 54,0; 54,5; 55,0; 100,0 ГГц. Антенны приемника располагаются на сканирующей опоре, имеющей азимутальный и угломестный приводы с оптическими цифровыми датчиками углов. Антенная апертура приемника может быть перекрыта поглотителем, определяющим калибровочный режим измерения «абсолютно черного тела» и имеющим электромеханический привод. Приемник содержит встроенный калибровочный генератор шума. Выходы всех каналов приемника через аналоговый мультиплексор поступают на специально разработанный блок оценивания среднего значения случайного процесса [5]. Оцениватель представляет собой

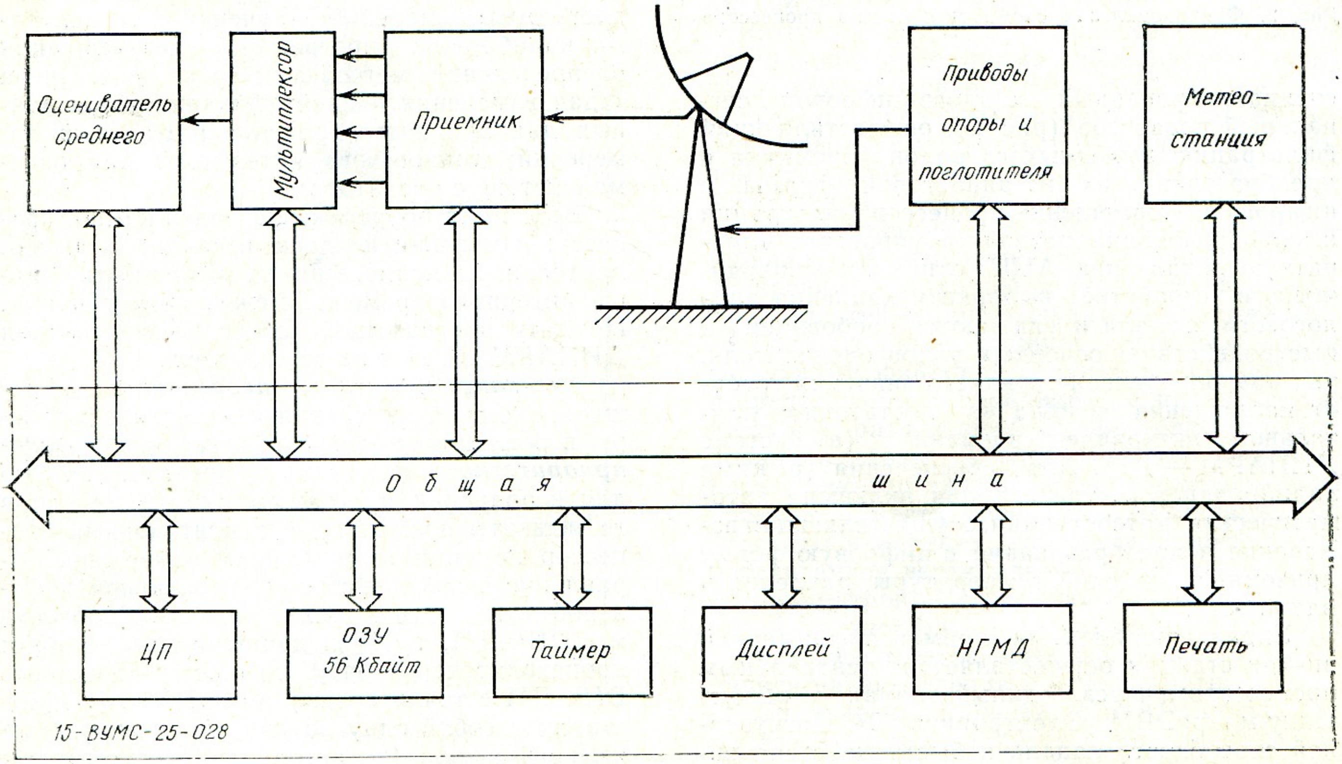


Рис. 1. Архитектура аппаратных средств станции ДИПАРМ

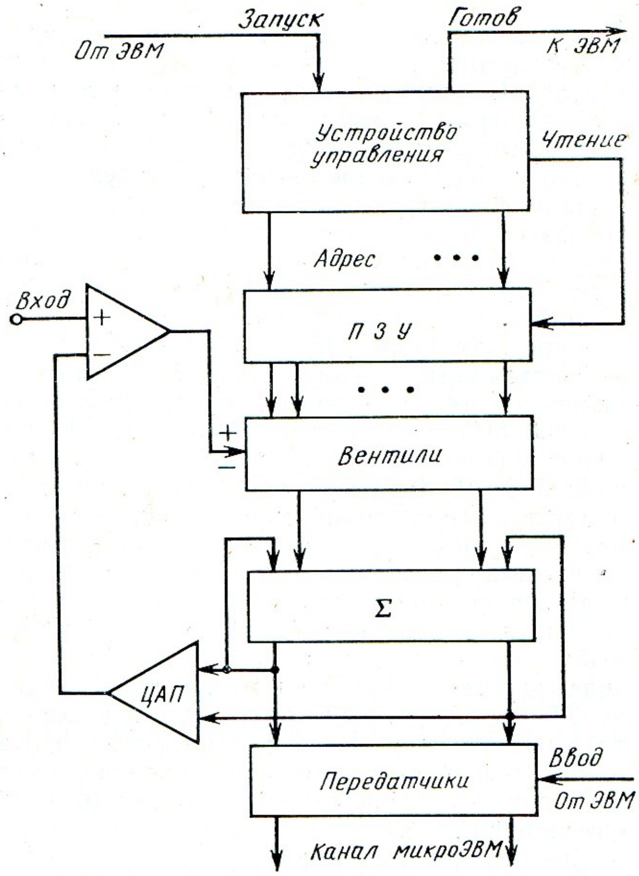


Рис. 2. Функциональная схема сигнального процессора

специализированный аналого-цифровой сигнальный процессор (рис. 2), осуществляющий фильтрацию выходных сигналов приемника и преобразование их из аналоговой формы в цифровую. Совмещение процессов фильтрации и преобразования позволяет упростить аппаратную реализацию АЦП: отпадает необходимость в устройстве выборки и хранения аналогового сигнала и снижаются требования к быстродействию основных узлов оценивателя, так как определение среднего значения требует наблюдения сигнала на достаточно длительном интервале времени (в системе ДИПАРМ — 1 с). Для обеспечения режима калибровки в состав станции включена автоматическая метеостанция, осуществляющая измерение и преобразование в цифровую форму приземных значений температуры, давления и влажности.

Управление блоками и обмен информацией внутри станции осуществляется центральным процессором через единый канал — общую шину микроЭВМ «Электроника 60». Нагрузочной способности канала и быстродействия общей шины вполне достаточно для обслуживания сравнительно небольшого числа устройств, входящих в станцию; обмен данными с центральным процессором для всех устройств (стандартных и нестандартных) становится унифицированным; асинхронность шины позволяет вести обмен в оптимальном темпе с устройствами различного быстродействия. Каждый блок, подключенный к общей шине, содержит в своем составе контроллер, реализующий стандартные протоколы обмена: «чтение», «запись», «прерывание».

Программные средства

Программное обеспечение станции ДИПАРМ состоит из минимальной версии ОС RAFOS и специальной программы DIPARM.

Версия ОС RAFOS включает следующие программы:

драйверы дисплея, устройства печати, накопителя на гибких магнитных дисках;

диспетчер запросов ввода-вывода на стандартные ВУ;

контроллер файловой структуры на гибких дисках;

контроллер системного времени;

диспетчер по загрузке программ с оверлейной структурой.

Программа DIPARM управляет всеми процессами измерения параметров и состоит из программы измерения излучения с процедурами калибровки; программы идентификации распределения метеопараметров; диспетчера задач измерения и идентификации; базы данных для архивного хранения результатов измерений; диалогового интерфейса для взаимодействия с оператором.

Весь цикл обработки разделен на два процесса: измерения и идентификации, так как длительность полного цикла может быть больше интервала времени между измерениями. Поэтому программное обеспечение станции ДИПАРМ состоит из двух разделов:

высокоприоритетного оперативного, где решается задача измерения излучения атмосферы в реальном масштабе времени, и ***низкоприоритетного фонового,*** предназначенного для исполнения программы идентификации в те моменты времени, когда центральный процессор не занимается задачей измерения. Управление взаимодействием оперативной и фоновой задач осуществляет диспетчер программы DIPARM, а обмен данными между ними происходит через базу данных. Программа DIPARM с точки зрения ОС RAFOS представляет собой одну задачу, написанную на языке Фортран (на ассемблере закодированы лишь драйверы нестандартных ВУ) и исполняющуюся в фоновом разделе, поэтому в па-

яти можно хранить лишь одну копию кода тандартных подпрограмм.

Программа оперативного раздела оформляется как подпрограмма на языке Фортран. Программа фонового раздела разбивается на ять подпрограмм, выполняющихся строго оследовательно и отражающих отдельные азы обработки. Время выполнения каждой одпрограммы должно быть существенно еньше минимального интервала между инициациями оперативной задачи. Каждой подпрограмме должна быть поставлена в соответствие функция, определяющая время выполнения в зависимости от режима идентификации. Погрешность определения времени выполнения определяет погрешность запуска оперативной задачи в запланированное время.

Часть программы DIPARM, представляющая собой собственно диспетчер, изображена на рис. 3. Обращение к нему (точка A) происходит в начале работы станции после установки режима измерения, по окончании процесса измерения и по окончании каждой фазы идентификации. Решение о том, какая

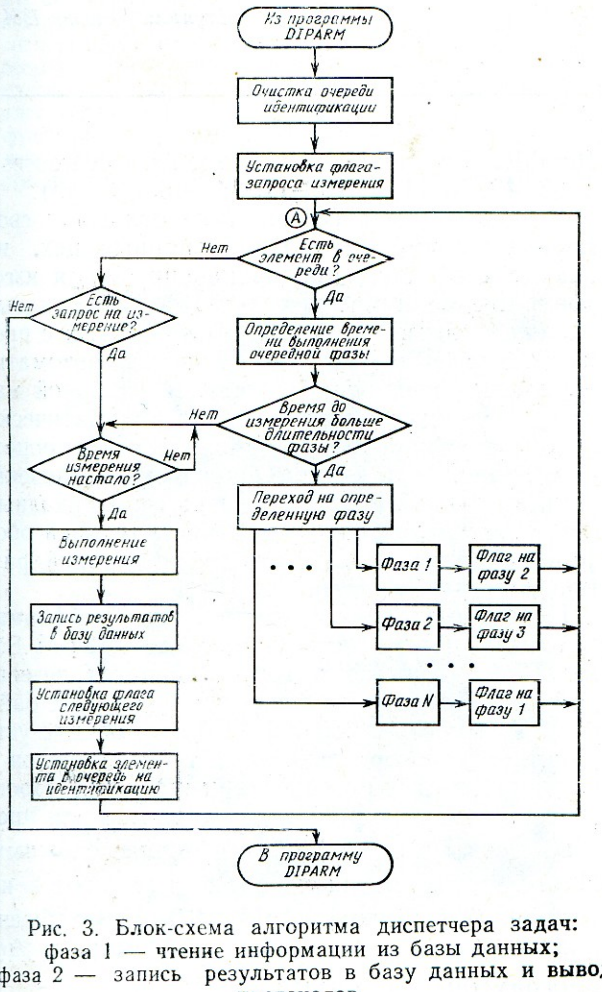


Рис. 3. Блок-схема алгоритма диспетчера задач: фаза 1 — чтение информации из базы данных; фаза 2 — запись результатов в базу данных и вывод протоколов

подпрограмма должна быть запущена, принимается на основе анализа очереди запросов на идентификацию, флага-запроса на измерение, флагов-указателей очередной фазы идентификации и текущего времени. Очередь запросов на идентификацию представляет собой список с организаций «первый вошел — первый вышел», элементами которого являются адреса базы данных, под которыми хранятся результаты необработанных измерений. Установку запроса в очередь осуществляет подпрограмма измерений. Флаг-запрос на измерение представляет собой астрономическое время, когда должен быть запущен очередной цикл измерений. Устанавливается флаг либо в начале работы системы, либо по окончании цикла измерений с учетом заданного интервала между циклами. Условием запуска оперативной задачи является совпадение текущего времени со временем, указанным в флаге-запросе. Условием запуска определенной фазы идентификации является наличие запроса в очереди на идентификацию, указание этой фазы в качестве следующей на выполнение. Время до очередного запуска оперативной задачи должно быть больше длительности фазы во избежание простоя центрального процессора. Кроме того, если время выполнения фазы предсказано с ошибкой в сторону уменьшения, то возможна задержка с запуском оперативной задачи. Недостатки такого диспетчера являются платой за его простоту и экономию памяти, достигаемую за счет хранения одной копии кода подпрограмм, используемых одновременно оперативной и фоновой задачами.

Таким образом, описанный диспетчер задач позволяет получать результаты в реальном масштабе времени, если длительность всего цикла определения пространственного распределения метеопараметров меньше интервала между измерениями, или проводить измерения с заданным интервалом, а результаты идентификации получать с задержкой.

База данных станции поддерживается средствами файловой структуры ОС RAFOS и представляет собой файл с произвольным доступом, разделенный на 466 записей фиксированной длины. Каждая запись содержит идентификатор записи в виде меток даты, времени и места проведения измерения, результаты измерения излучения атмосферы и рассчитанное вертикальное распределение метеопараметров. Первая запись представляет собой каталог базы, где отдельно отмечаются записи результатов измерения излучения и записи распределения метеопараметров.

Управление станцией ДИПАРМ осуществляется в режиме диалога с циклическим графом, реализованного в форме системы функциональных меню. Все параметры разбивают-

ся на классы и подклассы по функциональному единству. Выделяются следующие классы: режим измерения излучения, режим идентификации, режим калибровки приемника, задание метеорологической обстановки и т. п. Первый класс, в свою очередь, делится на подклассы: частоты для измерения, углы для измерения, последовательность измерений, интервал между измерениями и т. д. Вершинами графа являются функциональные меню, позволяющие выбрать конкретный класс (подкласс), а окончание графа соответствует отдельным параметрам станции. Таким образом, для изменения какого-либо параметра необходимо идентифицировать его путем ответа на ряд меню (в станции ДИПАРМ максимум на четыре), а затем задать для него новое значение. Для того, чтобы сократить число ответов оператора, каждая вершина графа, кроме первой, снабжена ветвью, соединяющей ее с вершиной предыдущего уровня. Наличие таких циклов позволяет вернуться к меню предыдущего уровня и выбрать другой параметр в том же подклассе (или другой подкласс в том же классе). Чтобы изменить несколько параметров в одном функциональном классе, нет необходимости повторять полностью весь диалог. Выход из режима диалога возможен из любой вершины.

Испытания станции показали, что она позволяет определять вертикальные распределения метеопараметров атмосферы в реальном масштабе времени с интервалом в 20 мин. Задержка в проведении измерений по отношению к заданному моменту при интервале между измерениями, равном 5 мин, не превышает 30 с. Среднее число ответов оператора за одну настройку равно восьми.

ЛИТЕРАТУРА

1. Чандрасекар С. Перенос лучистой энергии. — М., Иностранная литература, 1953.
2. Ленор В. Б. Косвенное определение температуры атмосферы с помощью измерений микроволнового излучения. — В кн.: Применение радиотеплолокации в метеорологии и океанологии. — Л., Гидрометеоиздат, 1969.
3. Кочетков А. А., Крылов В. В. — Автометрия, 1983, № 6. с. 11—17.
4. Китай Ш. Д., Рассадовский В. Н. — Радиофизика, 1981, 24, № 6, с. 680—687.
5. Корсаков С. Я., Крылов В. В. — Автоматика и вычислительная техника, 1981, № 1, с. 69—73.

Статья поступила 30 июня 1984 г.

РЖ ВИНИТИ

Гибкая автоматизация — современная тенденция развития производства. Flexible Automatisierung: im trend der Zeit. «Idee», 1983, № 10, (нем.).

Гибкое автоматизированное производство позволяет экономно обрабатывать детали как большими, так и небольшими сериями. В обрабатывающих центрах режущий инструмент может автоматически выбираться из накопителя и устанавливаться на станке. Отмечается большое значение сменных режущих пластин из твердых сплавов. Такие пластины могут также автоматически устанавливаться на инструменте.

Роботы обладают такой большой рабочей областью, что могут обслуживать несколько станков. Для большей оперативности целесообразно оборудовать роботы сменными захватами. Системы машинного конструирования позволяют учесть минимальные отклонения размеров, вызванные изменением внешних условий. Отдельные компоненты гибких автоматизированных систем не должны быть жестко связаны между собой. Конечной целью является возможность прямого диалога на всех уровнях системы.

Первый гибкий автоматизированный цех фирмы Citroën. Le mariage CFAO atelier. flexible. Peyrucal Jean Francois. «Mesures», 1983, **48**, № 11, 7, 10—11, 14—15, 18 (фр).

Фирма Citroën (Франция) открыла свой первый гибкий автоматизированный цех. на заводе в Медоне, предназначенный для изготовления механических деталей для новых моделей автомобилей, либо уже снятых с производства. Все машины полностью автоматизированы; перенос этих деталей от одной машины к другой осуществляется автоматически с помощью кареток. Управление всем процессом происходит без вмешательства человека. Однако ручными остаются некоторые входные операции: предварительная регулировка оборудования, загрузка и разгрузка деталей, разбор спорных случаев.

Централизованная управляющая информационная система состоит из двух ЭВМ Solar 16/65, одна из которых является резервной. ЭВМ снабжены блоком дисков 15М байт и ОЗУ емкостью 256 Кбайт, печатающим устройством со скоростью вывода 300 символов в секунду и двумя видеотерминалами. Программы разработаны на языке Promo для простых деталей и на APT для более сложных. После изготовления детали измеряются на трехмерной машине АМТ 130808, снабженной числовым управлением CNC Solar 16/40, программным обеспечением Promosur.

УДК 681.32—181.4:621.31

В. Д. Ковалев, А. К. Хансуваров, А. Т. Шевченко

МИКРОПРОЦЕССОРНАЯ СИСТЕМА ПРОТИВОАВАРИЙНОГО УПРАВЛЕНИЯ ПОВЫШЕННОЙ НАДЕЖНОСТИ

Средства автоматического управления электроэнергетическими системами обеспечивают требуемое качество электроэнергии по частоте и напряжению, экономичность работы энергосистем и поддержание в допустимых пределах основных параметров режима электротехнического оборудования. При повреждениях электротехнического оборудования устройства автоматики призваны обеспечивать надежность электроснабжения потребителей и живучесть энергосистем [1].

Для ликвидации каскадных аварий в объединенных энергосистемах важна локализация повреждений средствами *противоаварийной автоматики.* Система противоаварийной автоматики (ПАА) на базе МСУВТ В7 [2] вычисляет *управляющие воздействия* (УВ) для сохранения *устойчивости электростанций* при *аварийных возмущениях* и распределяет вычисленные и полученные УВ от устройств противоаварийного управления высшего уровня между агрегатами электростанции. ПАА — система программного типа и в нормальном режиме работы циклически опрашивает датчики аналоговых сигналов, анализирует топологическую схему сети, вычисляет управляющие воздействия по полиномиальным функциональным зависимостям от контролируемых режимных параметров, принимает задания внеплановой составляющей нагрузки от системного устройства автоматического регулирования частоты и мощности (*АРЧМ*) и распределяет его между агрегатами электростанции с учетом заданных весовых коэффициентов и технологических ограничений. При появлении сигналов прерывания от пусковых органов запускается блок опроса датчиков, фиксирующих интенсивность возмущения, и распределяются выбранные управляющие воздействия между агрегатами электростанций. ПАА принимает и распределяет также сигналы от устройств управления высшего уровня и передает им информацию о текущем режиме.

Входная информация ПАА — дискретные и аналоговые сигналы, характеризующие включенное состояние элементов системообразующей сети, значения режимных параметров, готовность агрегатов к участию в нормальном и противоаварийном управлении, значения кратковременных разгрузок и изменение нагрузок электростанций. Управляющие воздействия формируются в виде сигналов на отключение агрегатов и изменение их нагрузок. Всего система имеет 286 входных и 108 выходных аналоговых и дискретных сигналов.

К системе ПАА, работающей в ждущем режиме, предъявляются повышенные требования по надежности из-за большой ответственности выполняемых функций, обеспечивающих сохранение устойчивости энергосистем. Готовность ПАА должна быть не менее 0,997, частота ложных срабатываний — не более 0,01 в год. Большое количество входных и выходных сигналов практически исключает создание аппаратной избыточности в комплекте. Для обеспечения требуемых показателей надежности система ПАА выполняется из трех идентичных комплектов (рис. 1, 2) работающих *в режиме троирования* с использованием мажоритарной схемы контроля.

В состав каждого комплекта входят: входные и выходные устройства связи с объектом (УСО), микросистема из набора МСУВТ В7, индивидуальный субблок контроля (СКИ) и общий субблок контроля (СКО).

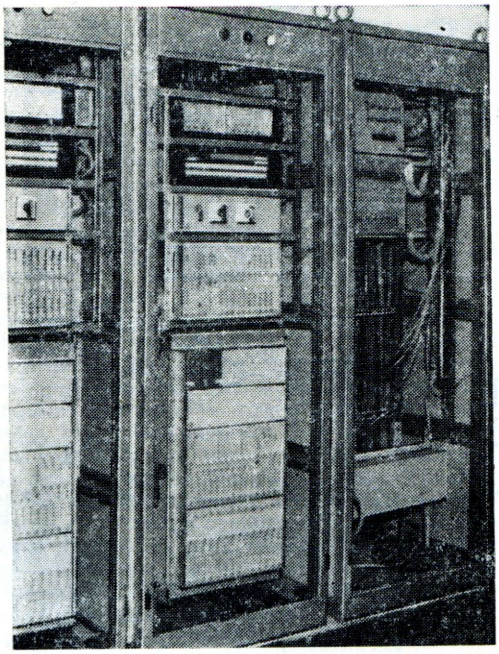


Рис. 1. Система противоаварийной автоматики

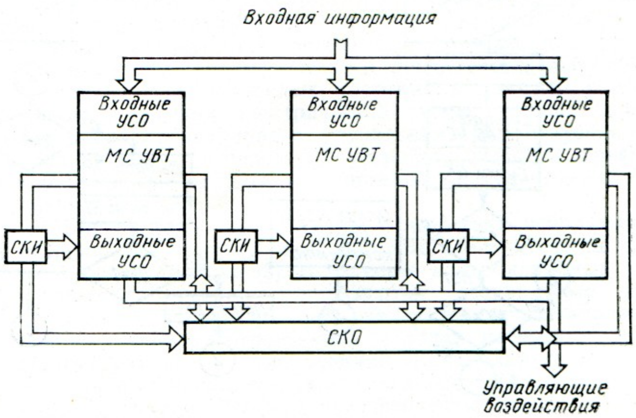


Рис. 2. Структурная схема системы ПАА

Одноплатная микроЭВМ типа ПМВ01, входящая в микросхему, содержит программируемые параллельные и последовательные интерфейсы, контроллеры прерывания, программируемые интервальные таймеры, ОЗУ и ППЗУ. Кроме того, микросхема из МСУВТ содержит две платы аналого-цифровых (АЦП) и цифро-аналоговых (ЦАП) преобразователей АВВ01, плату специального математического процессора ПСВ01, две платы расширения памяти МВ01 и пульт управления типа КЛ03. УСО представляет собой устройства нормализации сигналов и гальванической развязки на 1500 В. *Гальванические развязки* для дискретных сигналов выполнены с использованием герконовых реле, а для аналоговых сигналов — трансформаторных схем. СКИ управляет схемой включения выходных УСО. СКО сравнивают данные, выдаваемые комплектами в контрольных точках, и формируют импульсы для синхронизации их работы.

Обеспечение надежности системы

Исправность каждого комплекта проверяется в нормальном режиме, в режиме вычисления и распределения управляющих воздействий при получении сигналов от пусковых органов. Программно-аппаратные средства обеспечения надежности ПАА организованы так, что в доаварийном режиме решаются задачи выявления отказавших комплектов, переключения на резервный комплект и диагностики отказавших элементов в неисправном комплекте вплоть до элемента замены. В ПАА применен принцип функциональной проверки в контрольных точках программы. На мажоритарную схему контроля каждым комплектом выдаются сигналы, пропорциональные значениям, характеризующим результат проверки плат устройств связи с объектом и МСУВТ В7. Контрольные точки выбраны так, что уход комплектов по различным программным ветвям может быть только между контрольными точками при значениях параметров, близких к граничным или к значениям условных переходов. При этом не требуется межмашинный обмен. Это существенно повышает надежность устройства.

Комплект диагностируется тремя видами проверок: входные УСО проверяются сравнением в СКО информации, считываемой с УСО; выходные УСО контролируются сравнением выданной величины с ее значением, принятым в микроЭВМ по обратной связи; МСУВТ В7 контролируются программными и аппаратными средствами. Для диагностики АЦП программно выдается фиксированное значение с ЦАП и через один из каналов АЦП принимается это же значение. Математический процессор диагностируется тестовой программой. Для проверки микроЭВМ управляющие воздействия, вычисленные в нормальном режиме работы, подаются на схему сравнения СКО. В режиме противоаварийного уп-

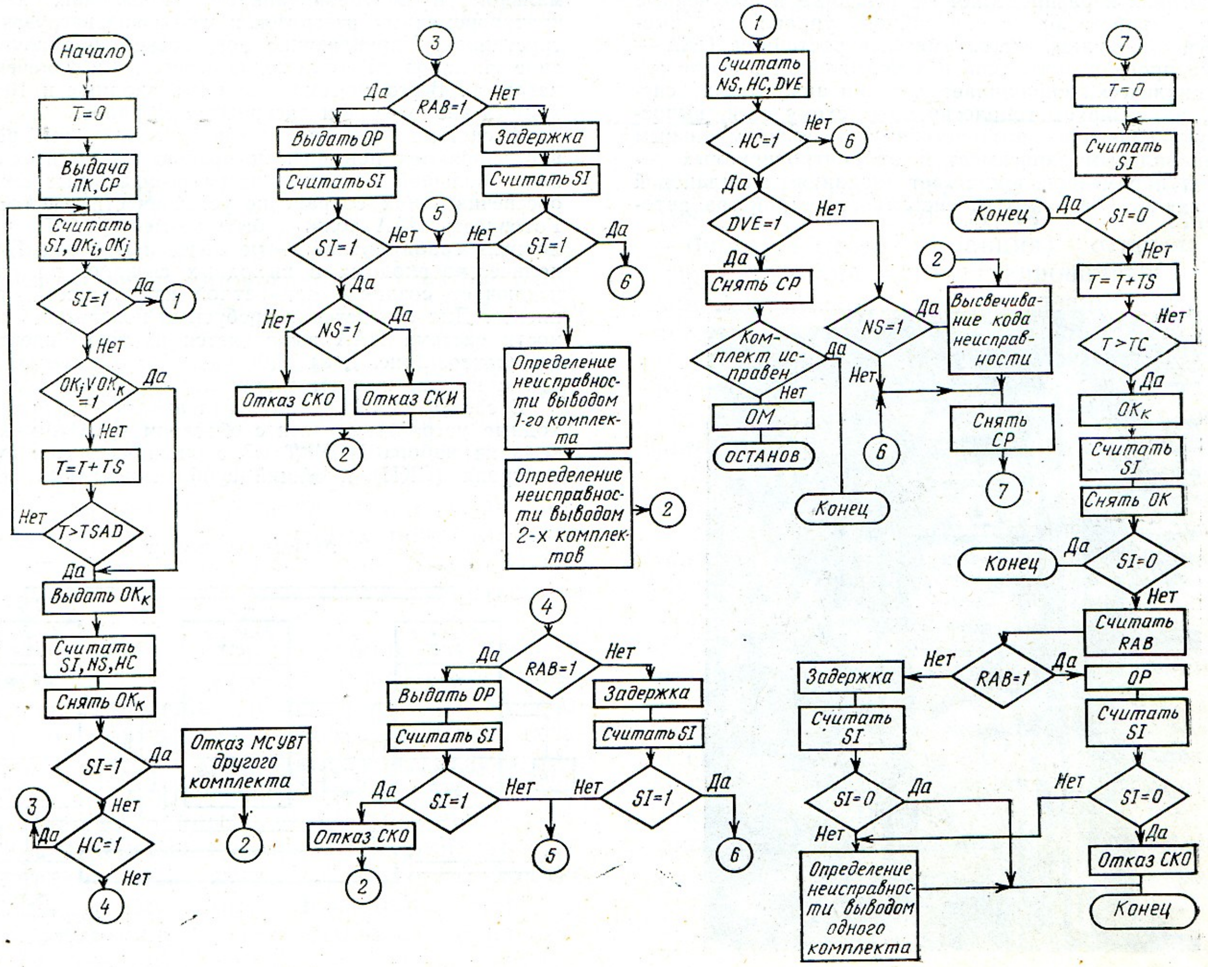


Рис. 3. Блок-схема алгоритма отыскания неисправностей в системе ПАА

равления вычисленные УВ поступают на схему сравнения при исправности трех комплектов. В противном случае рабочий комплект выдает УВ на исполнительные органы без проверки.

Каждый комплект может находиться в одном из состояний: работа, резерв, неисправность, ремонт. Текущее состояние указано на лицевой стороне шкафа комплекта. В нормальном режиме все комплекты участвуют в определении неисправностей (трехкомплектная система). В работе используется СКО одного комплекта, выходные УСО которого включены. При обнаружении неисправности в рабочем комплекте СКИ отключает выходные УСО, затем включаются УСО следующего по приоритетности комплекта.

Отказы элементов комплекта (контролируемые повреждения) выявляются программными и аппаратными средствами в ПАА и индицируются на пульте управления КЛОЗ микросистемы. Аппаратные сигналы отказа МСУВТ высвечиваются на светодиодах СКИ. Неконтролируемые повреждения двух и более независимых элементов, выявление отказов которых в течение цикла расчета не предусмотрено алгоритмом, определяются поочередным автоматическим выводом из сравнения комплектов ПАА.

Каждый комплект контролирует выходные УСО и участвует в выявлении неисправного элемента. При этом в СКИ поступают следующие сигналы:

— отказ МСУВТ (ОМ). В его состав входят три сигнала (обращение к несуществующей памяти или устройствам ввода-вывода, нарушение четности во время обращения к памяти, неисправность системы питания), формируемые аппаратно, и один сигнал, формируемый программно при неисправности плат МСУВТ.

— отказ УСО (ОР);

— отказ комплекта (ОК).

По сигналу ОМ комплект выводится в состояние РЕМОНТ, а ПАА — в режим двухкомплектной системы. По сигналу ОР СКИ рабочего комплекта отключает выходные УСО. Комбинационная схема СКО формирует команду на включение следующего по приоритетности комплекта, которая реализуется в СКИ включением выходных УСО. При наличии сигнала ОР комплект участвует в отыскании неисправностей. В случае отказа микроЭВМ соответствующий комплект отключается по сигналам ОК, выдаваемыми исправными комплектами на логическую схему отключения неисправного комплекта. По двум сигналам ОК от исправных комплектов снимается питание с неисправного.

При контроле исправности входных УСО и МСУВТ информация, считываемая с соответствующего элемента или определенная в результате расчетов УВ, подается на схему сравнения СКО. Выдача соответствующей информации осуществляется в контрольных точках (КТ) цикла доаварийного расчета. В случае выявления неисправности на индикаторе пульта управления КЛОЗ высвечивается номер КТ.

Блок-схема алгоритма отыскания неисправностей в системе ПАА представлена на рис. 3. МикроЭВМ каждого комплекта выдает на схему сравнения СКО значение контрольного параметра (ПК), определяемого в очередной контрольной точке, и сигнал разрешения сравнения (СР). При наличии трех сигналов СР и совпадении сравниваемых величин схема сравнения СКО вырабатывает синхроимпульс (SI). При поступлении сигнала SI микроЭВМ каждого комплекта снимает сигнал СР. После этого схема сравнения гасит синхроимпульс, разрешая микроЭВМ всех комплектов дальнейшую обработку программы. В случае отклонения одной из сравниваемых величин и при наличии трех сигналов СР схема сравнения вырабатывает сигнал общей неисправности (НС), поступающий на все комплекты, и сигнал индивидуальной неисправности NS, который подается на СКИ и МСУВТ неисправного комплекта. По сигналу NS СКИ рабочего комплекта отключает выходные УСО и переводит комплект в состояние НЕИСПРАВНОСТЬ. У резервного комплекта по сигналу NS вырабатывается запрет на включение выходных УСО. В работу включается следующий по приоритетности исправный комплект, у которого формируется сигнал SI, поступающий на МСУВТ всех комплектов.

При обработке информации в доаварийном цикле из-за неидентичности частот тактовых генераторов возможно расхождение в выполнении команд каждым комплектом приблизительно на 1200 мкс. Для синхронизации работы комплектов введены задержки TSAD-1200 мкс и ТС-100 мкс. Для снижения времени взаимного расхождения микроЭВМ проводится считывание и анализ сигналов ОК других комплектов в дополнительном цикле. Максимальное расхождение работы комплектов после дополнительного цикла не превышает 40 мкс.

Отсутствие в режиме сравнения сигнала SI может быть в случае неисправности одной из микроЭВМ (отказ процессора или параллельного интерфейса, через который выдается сигнал СР), СКО или СКИ.

При наличии двух сигналов ОК от исправных комплектов и отсутствии сигнала СР от неисправного комплекта (на выходе программы при сигнале СР) у последнего логической схемой отключается питание. Код неисправности индицируют другие комплекты. При неисправности СКО или СКИ комплект отключается сигналом ОР. Если возникли два и более неконтролируемых отказа, неисправные комплекты определяются поочередным выводом из сравнения.

Доаварийный режим двухкомплектной системы отличается от режима трехкомплектной тем, что СКО вырабатывает сигнал DVE, а SI появляется и снимается с выдачей и снятием двух сигналов СР от исправных комплектов. При отклонении одной из сравниваемых величин в двухкомплектной системе сигнал NS не вырабатывается, а сигналы НС и DVE подаются на оба комплекта. По этим сигналам в комплектах по специальной программе определяются неисправности сравнением текущего значения в контрольной точке с соответствующим значением, записанном в предыдущем цикле. Неисправный комплект останавливается. Устройство ПАА переводится в режим однокомплектной системы. Если работает один комплект, сигналы NS и НС не образуются, а сигнал SI появляется и снимается соответственно при выдаче и снятии одного сигнала СР.

Сбои в работе системы устраняются однократным автоматическим перезапуском всех комплектов при определении соответствующих неисправностей.

Функциональные алгоритмы

Алгоритм противоаварийного управления для обеспечения устойчивости параллельной работы электростанций реализуется в системе с учетом каскадного развития аварии, автоматического повторного включения элементов сети, затяжных коротких замыканий и возможного деления электрической схемы станции. В режиме противоаварийного управления ПАА запускается управляющим сигналом от устройств высшего уровня иерархии или от пусковых органов противоаварийной автоматики.

Для расчета управляющих воздействий используются полиномиальные функциональные зависимости не выше второго порядка [3]. В качестве контролируемых режимных параметров используются доаварийные нагрузки генераторов и линий электропередач, напряжения на шинах электростанции и узлах нагрузки, аварийные изменения этих величин. Контролируемые режимные параметры поступают в систему ПАА в виде токовых сигналов. Весовые коэффициенты полиномов зависят от схемы энергосистемы. Матрица весовых коэффициентов, строки которой соответствуют ремонтным схемам, рассчитываются предварительно и записываются в ППЗУ. Информация об отключении элементов се-

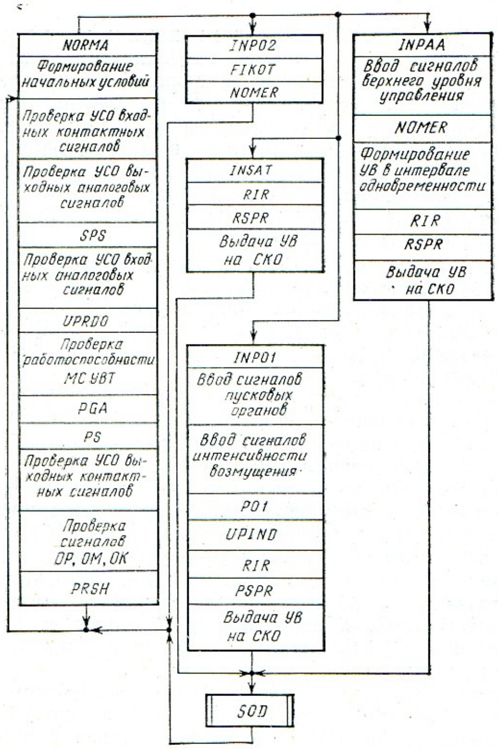


Рис. 4. Блок-схема алгоритма функционирования системы ПАА

ти поступает в контактной форме. Функциональные зависимости формируются для всех пусковых органов, при действии которых возможно нарушение устойчивости параллельной работы электростанций.

Распределение вычисленных УВ осуществляется равномерно между агрегатами с учетом регулировочного диапазона. При исчерпании суммарного регулировочного диапазона генераторы отключаются.

Блок-схема алгоритма функционирования системы ПАА представлена на рис. 4. Программа NORMA выполняет подготовительные операции для противоаварийного управления и диагностирует неисправности. Подпрограмма SPS вводит информацию о режимных параметрах. Подпрограмма UPRDO рассчитывает значения управляющих воздействий по полиномиальным функциональным зависимостям [3] текущего состояния сети для всех пусковых органов и ступеней интенсивности возмущений. Формирует информацию о готовности каждого агрегата к изменению нагрузки в режимах нормального регулирования и противоаварийного управления подпрограмма PGA. В целях оптимизации вычислений в режиме противоаварийного управления в подпрограмме PS ранжируются элементы массивов регулировочных диапазонов агрегатов электростанции. Распределяет внеплановую составляющую нагрузки подпрограмма PRSH.

Программа INPO1 запускается при повреждениях в схеме выдачи мощности по прерыванию. Определяется номер пускового органа и вводится информация об интенсивности возмущения. Подпрограмма PO1 рассчитывает значения УВ для сохранения устойчивости с учетом возможности затяжного короткого замыкания. При попадании пускового сигнала в заданный интервал одновременности (формируется подпрограммой SOD) корректируются вычисленные УВ с учетом предыдущего УВ (подпрограмма UPIND). Подпрограмма RIR распределяет УВ на кратковременную разгрузку и отключение агрегатов. Управляющие воздействия для обеспечения статической устойчивости послеаварийных режимов распределяются в подпрограмме RSPR. Для контроля исправности рабочего комплекта перед реализацией УВ выдается информация на блок контроля.

Программа INPO2 запускается при отключении элементов сети по прерыванию. Она фиксирует отключенный элемент сети (подпрограмма FIKOT) и формирует новый номер состояния сети (подпрограмма NOMER). Программа INSAT осуществляет распределение УВ, вычисленных для затяжных коротких замыканий в программе INPO1, между агрегатами. Программа INPAA обрабатывает информацию об УВ, получаемых с верхнего уровня противоаварийного управления, с учетом деления электрической схемы станции и попадания в интервал одновременности.

Цикл работы программы NORMA не превышает 5 с, а время отработки программ, запускаемых по прерыванию (INPO1, INPO2, INSAT, INPAA) — 30 мс. Общий объем программного обеспечения в ППЗУ — 20 Кбайт, дополнительный объем ППЗУ для весовых коэффициентов — 28 Кбайт. Программное обеспечение разработано на языке ассемблер с использованием принципов структурного программирования.

Конструктивное исполнение

Система ПАА выполнена в четырех шкафах (см. рис. 1) с габаритными размерами 800×800×2200 мм. Каждый комплект занимает один шкаф. Четвертый, кроссовый шкаф, служит для связи ПАА с внешними устройствами. Он имеет панель управления, на которой установлены: переключатели очередности отключения агрегатов в режиме противоаварийного управления; переключатели участия агрегатов в режиме противоаварийного управления и нормального регулирования нагрузки; кнопка начального пуска; кнопка разрешения отработки задания внеплановой составляющей нагрузки от устройств верхнего уровня управления. Пульт управления микросистемы КЛОЗ используется для запуска тестовых программ, автоматической индикации номера неисправного элемента, индикации (по вызову) результатов работы ПАА в режиме противоаварийного управления.

Система бесперебойного питания и устройства связи с объектом (для приема и выдачи дискретных сигналов), обеспечивающие гальваническую развязку 1500 В, размещены на передней и задней поворотных рамах шкафа. Система бесперебойного питания ПАА подключается к сети переменного (220 В, 50 Гц) и постоянного (220 В) тока.

Микропроцессорная система ПАА для противоаварийной автоматики энергосистем не содержит в своем составе специализированных аппаратных узлов, и поэтому применима для широкого класса АСУ ТП. Система ПАА адаптируется к управлению конкретными объектами лишь с помощью соответствующего функционального программного обеспечения.

ЛИТЕРАТУРА

1. Бортник И. М., Ковалев В. Д., Корнилов Б. А. Системы автоматического управления электротехническим оборудованием для энергетики с применением микроЭВМ. — Электротехника, 1982, № 6.

2. Шереметьевский Н. Н., Долкарт В. М. Магистрально-модульные микросредства управляющей вычислительной техники. — Микропроцессорные средства и системы, 1984, № 2, с. 24—27.

3. Ковалев В. Д. Алгоритмы управляющих воздействий противоаварийной автоматики электроэнергетических систем. — Электричество, 1981, № 12, с. 13—19.

Статья поступила 18 апреля 1984 г.

УДК 681.335

Н. Т. Абдуллаев, Л. З. Измайлова, Э. А. Тургиев

СОПРЯЖЕНИЕ АНАЛОГО-ЦИФРОВОГО ПРЕОБРАЗОВАТЕЛЯ С ТАЙМЕРОМ В МИКРОПРОЦЕССОРНОЙ СИСТЕМЕ

Ввод информации, представленной в аналоговой форме, с помощью аналого-цифрового преобразователя (АЦП) в микропроцессорную (МП) систему и ее последующая обработка требуют определенных средств сопряжения. На них возлагаются функции регулирования частоты запуска или опроса АЦП, подсчет заданного числа преобразований, режим и порядок опроса каналов (в многоканальных системах) и другие. В МП системах сбора данных, работающих в реальном времени, сопряжение с АЦП целесообразно осуществлять с помощью программируемого таймера, который позволяет реализовать указанные функции с большим быстродействием по сравнению с программным способом.

Таймер КР580ВИ53 микропроцессорного комплекта БИС серии КР580 состоит из трех независимых 16-разрядных счетчиков и управляющего регистра [1]. В регистр управления таймера по шине данных (ШД) записывается код, определяющий номер счетчика, режим его работы, систему счисления, запись и считывание одного или двух байтов содержимого счетчика.

Схема сопряжения АЦП с таймером (ТМР) в МП системе (см. рис.) реализует функции регулирования частоты запуска АЦП и подсчет заданного числа преобразований. Регулирование частоты запуска АЦП осуществляется с помощью счетчика Сч0 таймера. На вход C_0 счетчика подаются тактовые сигналы с ча-

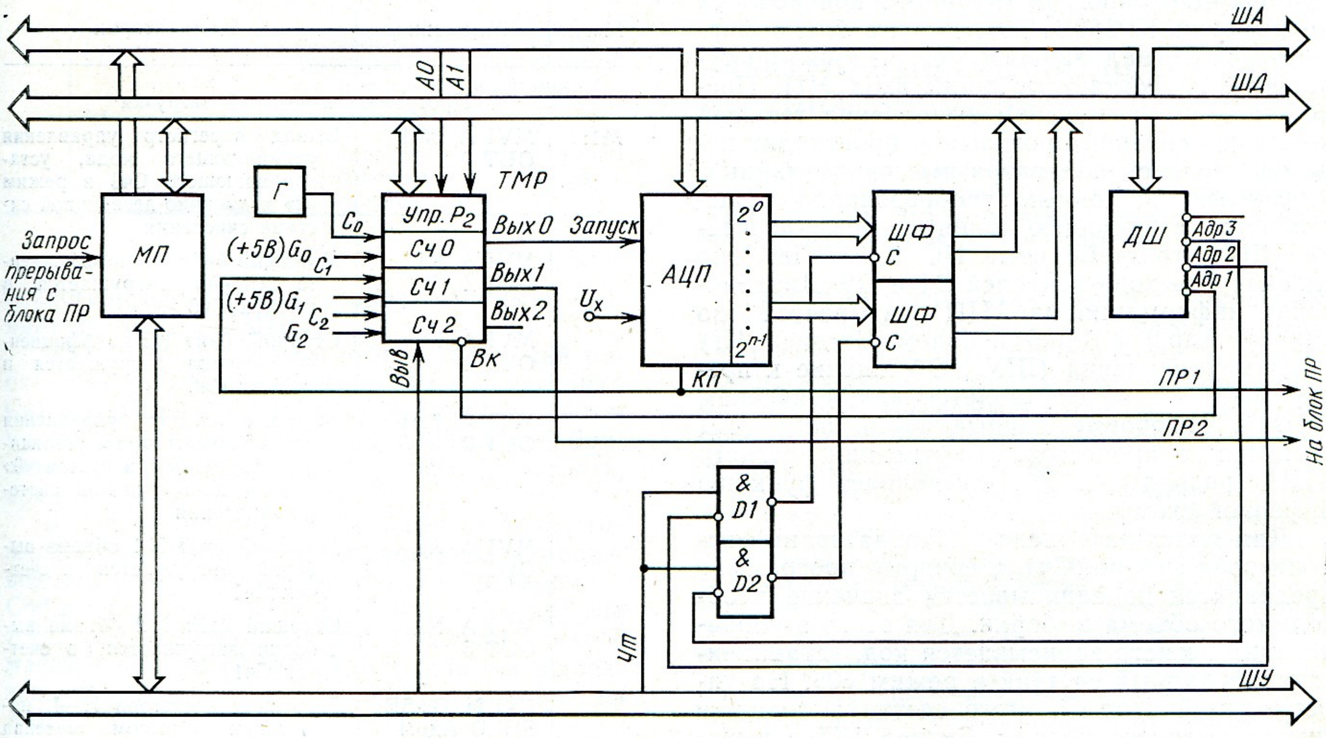


Схема сопряжения АЦП с таймером

стотой $f_т \leqslant 2$ МГц от внешнего генератора (Г), а на управляющий вход G_0 — сигнал высокого уровня.

Для запуска АЦП выход счетчика Вых.0 соединяется со входом «Запуск» преобразователя. Частота запуска АЦП определяется коэффициентом деления $K_д$, значение которого записывается в счетчик программным способом. В управляющий регистр записывается код, устанавливающий данный счетчик в режим деления частоты (режим «2»). В результате этого частота импульсов на выходе счетчика устанавливается равной $f_т/K_д$. Для счетчика не требуется перезагрузка в процессе работы. Регулирование частоты запуска АЦП осуществляется загрузкой счетчика новым значением коэффициента деления. Операция загрузки счетчика не влияет на текущий период импульсов на выходе счетчика и новое значение $K_д$ определит продолжительность следующих периодов.

Управляющий сигнал G может быть использован для синхронизации счетчика от любого внешнего устройства. Считывание результатов преобразования и их ввод из АЦП в процессор организуется по методу передачи данных с прерыванием программы. По окончании цикла преобразования АЦП формирует сигнал «Конец преобразования» (КП), который в качестве сигнала прерывания ПР1 подается на блок прерывания «Блок ПР» (например, БИС программируемого контроллера прерываний КР580ВН59). В этом блоке формируется сигнал запроса прерывания, поступающий на соответствующий вход МП. После этого, по завершении выполнения текущей команды основной программы, происходит переход к адресу подпрограммы ввода данных.

Считывание данных преобразования осуществляется с помощью шинных формирователей ШФ (тристабильные ИС К155ЛП8) по сигналу с выхода вентилей D1 и D2. Двухбайтовая информация из АЦП считывается по адресам Адр.2 и Адр.3 сигналом «Чтение» (Чт) с шины управления (ШУ). Обращение к преобразователю осуществляется как к ячейкам памяти. Цифровые данные через шину ШД вводятся в процессор. Внутренний регистр АЦП (разряды $2^0 \ldots 2^{n-1}$) выполняет функцию буферной схемы.

Для подсчета числа преобразований использован счетчик Сч1, в который программно предварительно записывается значение необходимого объема выборки. Для этого в управляющий регистр записывается код, устанавливающий данный счетчик в режим «0». На управляющий вход G_1 этого счетчика подается сигнал высокого уровня. Сигнал КП с выхода АЦП поступает на вход C_1 счетчика 1. По достижении заданного числа преобразований на выходе Вых.1 счетчика возникает высокий уровень, используемый как сигнал запроса прерывания ПР2, информирующий об окончании выборки заданного объема. Для исключения неправильной работы схемы при одновременном действии обоих сигналов прерывания сигнал ПР1 должен иметь более высокий приоритет по сравнению с сигналом ПР2.

Обычно таймер подключается к шинам МП системы как внешнее устройство (см. рисунок), однако к управляющему регистру и счетчикам таймера возможно обращение как к ячейкам памяти. Обращение к таймеру осуществляется через адресные входы А0, А1 и вход ВК («Выбор кристалла») с помощью управляющего сигнала «Вывод» с шины ШУ. При наличии низкого потенциала на входе ВК, связанном с выходом Адр1 дешифратора (ДШ), доступ к управляющему регистру происходит при А0=А1=1. к счетчику Сч0 — при А0=А1=0, к Сч1 — при А0=1, А1=0, к Сч2 — при А0=0, А1=1. Низкий уровень на выходе Адр.1 дешифратора ДШ должен формироваться при каждом обращении к счетчикам и управляющему регистру таймера.

По приведенной схеме можно осуществлять сопряжение таймера с различными типами АЦП, имеющими выходной сигнал «Конец преобразования» (например, АЦП К572ПВ1) или

Таблица

Метка	Мнемоника	Комментарии
		Программа начальной загрузки
М1:	MVI A, 35H OUT 7	Вывод в регистр управления управляющего кода, устанавливающего Сч0 в режим «2» в двоично-десятичной системе счисления
	MVI A, K1 OUT 4	Младший байт К1 коэффициента деления загружается в счетчик Сч0
	MVI A, K2 OUT 4	Старший байт К2 коэффициента деления загружается в счетчик Сч0
	MVI A, 71H OUT 7	Вывод в регистр управления управляющего кода, устанавливающего Сч1 в режим «0» в двоично-десятичной системе счисления
	MVI A, N1 OUT 5	Младший байт N 1 объема выборки загружается в счетчик Сч1
	MVI A, N2 OUT 5	Старший байт N 2 объема выборки загружается в счетчик Сч1
	LXI H, BASE SHLD Адр.4	Загрузка регистров H—L начальным адресом массива памяти для записи данных из АЦП и запоминание его по Адр.4, Адр.5

Продолжение табл.

Метка	Мнемоника	Комментарии
	Основная программа	
	RST 2	Прерывание МП по сигналу КП
	PUSH PSW PUSH H	Сохранение в стеке содержимого аккумулятора, регистра состояния и регистров H—L
	JMP M2	Переход к подпрограмме ввода данных из АЦП.
M2:	LHLD Адр.2	Ввод данных из АЦП в регистры H—L ([Адр.2] → L, [Адр.3] → H).
	XCHG	Пересылка данных из регистров H—L в регистры D—E
	LHLD Адр.4	Загрузка регистров H—L текущим адресом массива памяти
	MOV M, E	Пересылка в память первого байта данных;
	INX H	Инкремент адреса массива памяти
	MOV M, D	Пересылка в память второго байта данных.
	INX H	Инкремент адреса массива памяти
	SHLD Адр.4	Запоминание текущего адреса массива по Адр.4, Адр.5.
	POP H POP PSW	Восстановление содержимого регистров H—L, аккумулятора и регистра состояния
	EI	Разрешение прерывания
	RET	Восстановление содержимого счетчика команд и возврат к основной программе.

аналогичный сигнал «Готовность данных» (например, АЦП К113ПВ1).

Фрагмент программы, обеспечивающий работу счетчика Сч0 в режиме «2» и счетчика Сч1 в режиме «0» в двоично-десятичной системе счисления, а также ввод данных преобразования в процессор и их запись в память, приведен в таблице.

В программе начальной загрузки программируются режимы работы счетчиков и производится загрузка коэффициента деления в Сч0 и значение объема выборки в Сч1. Кроме того, здесь же определяется начальный адрес массива памяти (Адр.4, Адр.5) для последовательной записи данных преобразования из АЦП.

При применении в приведенной схеме АЦП, не имеющего выходного сигнала «Конец преобразования», третий счетчик Сч2 таймера может быть использован в режиме «0» для формирования сигнала запроса прерывания ПР1 с задержкой по отношению к сигналу запуска на время несколько большее времени преобразования АЦП.

В многоканальных системах сбора информации [2] счетчик 2 таймера целесообразно использовать для управления по адресным входам аналоговым коммутатором, т. е. для программирования режима и порядка опроса каналов. Примером таких систем могут служить многоканальные системы измерения параметров датчиков.

ЛИТЕРАТУРА

1. Торгов Ю. И. Программируемый таймер КР580ВИ53 и его применение. — Микропроцессорные средства и системы, 1984, № 1, с. 77—84.

2. Малинин В. В., Шмид Г. Г. Микроинтерфейсы и системы сбора-распределения данных. Часть 1. Принципы сопряжения, программное обеспечение и стандартизация. — М.: ЦНИИ «Электроника», 1981. с. 38. — (Обзоры по электронной технике. Сер. 3 «Микроэлектроника», вып. 1 (768)).

Статья поступила 24 апреля 1984 г.

УДК 681.32

Г. В. Зеленко, В. В. Панов, С. Н. Попов

ЭЛЕКТРОННЫЙ «КВАЗИДИСК» ДЛЯ ПЕРСОНАЛЬНОЙ ЭВМ

В качестве внешней памяти микроЭВМ все чаще используют дополнительное ОЗУ большой емкости (имитаторы дисков). Это существенно снижает стоимость микроЭВМ и повышает ее быстродействие.

Авторами разработан образец микроЭВМ, для которой в качестве устройств ввода-вывода применяются бытовые приборы: телевизор с черно-белым изображением и кассетный магнитофон. МикроЭВМ была описана в журнале «Радио» [1]. Кратко напомним читателям ее основные характеристики.

МикроЭВМ с оперативной (62 Кбайт) и постоянной (2 Кбайт) памятью выполнена на базе микропроцессора КР580ИК80А. *Дисплей* на базе телевизора с черно-белым изображением отображает 32×64 символов; возможен псевдографический режим работы с разрешением 128×64 точек. Для уменьшения стоимости клавиатуры дисплея при ее выполнении использовано программное сканирование. Внешняя память — *бытовой кассетный магнитофон* без каких-либо переделок (скорость записи-считывания 1500 бод, плотность записи 32 бит/мм). Сопряжение с магнитофоном в основном программное. В состав микроЭВМ входят *программатор* для ППЗУ с ультрафиолетовым стиранием и *отладочный модуль* для отладки аппаратных средств.

Программное обеспечение включает в себя программы монитор (2 Кбайт), записанный в ППЗУ; редактор текстов; транслятор с ассемблера и интерпретатор языка Бейсик, вводимые с кассетного магнитофона.

Опыт эксплуатации микроЭВМ показал высокую достоверность и приемлемую скорость обмена информацией с магнитофоном. Это позволило модернизировать микроЭВМ для достижения эксплуатационных и сервисных возможностей, характерных для микроЭВМ с НГМД. Для этого в состав микроЭВМ был включен **«квазидиск» — дополнительное ОЗУ, имитирующее НГМД.**

К шинам микроЭВМ его можно подключить через программируемые периферийные адаптеры КР580ВВ55 для выдачи сигналов адресов, шину данных и сигналы управления. Схемное решение описано в работе [2] для микропроцессора Z80. **Недостаток** этого варианта — значительное снижение скорости передачи данных между дополнительной памятью и основной из-за необходимости применять команды ввода-вывода для формирования адресных и управляющих сигналов при пересылке каждого байта. Кроме того, данный способ предполагает, что дополнительная память способна работать асинхронно по отношению к центральному процессору.

Второй вариант подключения «квазидиска» к шинам микроЭВМ — использование команд пересылки по косвенному адресу (например, MOV M, A или MOV A, M). Перед выполнением этих команд в паре регистров H, L помещается адрес соответствующей ячейки в дополнительной памяти. Специальная схема позволяет считывать код операции из основной памяти и записывать (или считывать) операнд в дополнительную память в ходе выполнения команды. Так шины микроЭВМ переключаются во внутрисхемных эмуляторах. **Недостаток** этого способа — относительно большая аппаратурная избыточность.

В разработанной авторами микроЭВМ используется **более эффективный способ организации обмена информацией с «квазидиском».** При работе микропроцессора КР580ИК80А в начале каждого машинного цикла на шине данных выставляется слово состояния микропроцессора, указывающее на тип данного машинного цикла. При выполнении команд работы со стеком (PUSH R, POP R и др.) в слове состояния формируется соответствующий кодовый признак (СТК), который используется для формирования сигнала разрешения обращения к дополнительной памяти.

Существенное преимущество этого варианта — высокая скорость пересылки данных между «квазидиском» и основной памятью, так как при выполнении команд обращения к стеку пересылаются два байта при использовании стекового механизма адресации. Время «загрузки» программы объемом 32 Кбайт из дополнительной памяти в основную — 1,5 с.

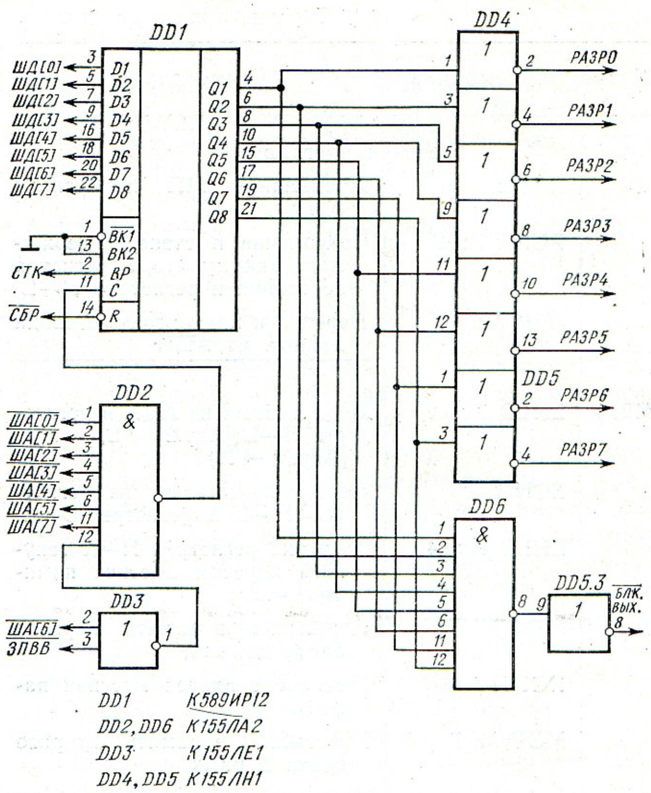


Схема модуля управления дополнительной памятью — «квазидиском»

В модуле управления дополнительной памятью (см. рисунок) многорежимный буферный регистр DD1 используется в качестве порта вывода и хранит признаки разрешения выбора блоков дополнительной памяти (до 64 Кбайт). При запуске микроЭВМ по сигналу ЗПВВ в этот порт записывается код FFH, запрещающий обращение к дополнительной памяти. Адрес порта — 40H. Выходные линии порта находятся в высокоимпедансном состоянии до прихода сигнала СТК на вход ВР микросхемы DD1. При этом сигналы разрешения обращения к блокам дополнительной памяти и блокировки основного ОЗУ (БЛК ВЫХ) отсутствуют.

После записи «Лог. 0» в какой-либо разряд микросхемы DD1 и при появлении сигнала СТК в регистре слова состояния микропроцессора на одном из выходов микросхемы DD1 также появляется сигнал «Лог. 0». Это вызывает формирование сигнала блокировки основного ОЗУ (БЛК ВЫХ) и сигнала разрешения обращения к соответствующему блоку дополнительной памяти РАЗР N. После завер-

шения операции обращения к дополнительной области памяти в микросхему DD1 вновь записывается код FFH.

В качестве дополнительного ОЗУ — «квазидиска» — в микроЭВМ используются модули ОЗУ (64 Кбайт), выполненные на микросхемах К565РУ3А [1]. При этом выход БЛК ВЫХ модуля управления связывается с входом БЛК ВХ модуля основной памяти, а выходы РАЗР N — с входами БЛК ВХ соответствующих блоков дополнительной памяти. Один из модулей дополнительной памяти выполнен на базе микросхем РПЗУ К573РФ2 и служит для хранения собственно операционной системы и наиболее часто используемых системных программ (экранный редактор текста, отладчик, ассемблер и др.).

Каждый из блоков дополнительной памяти имитирует НГМД со следующими информационными характеристиками:

общая информационная емкость . .	64 Кбайт
число «дорожек»	64
число «секторов» на дорожке . .	8
информационная емкость «сектора» .	128 байт

Между основной памятью и имитатором НГМД данные пересылаются при помощи подпрограмм «Чтение сектора» (табл. 1) и «Запись сектора» (табл. 2). Эти подпрограммы в свою очередь обращаются к подпрограмме «Расчет адреса», которая по заданному номеру «дорожки» и «сектора» определяет физический адрес информации в имитаторе НГМД и помещает его в пару ячеек PHAD.

Перед началом работы микроЭВМ с магнитофона в дополнительную память загружаются необходимые программы (и на ленте, и в дополнительной памяти они хранятся в виде поименованных файлов). Все манипуляции с файлами производятся при помощи директив операционной системы. Перед окончанием работы полученные результаты (тексты программ, программы, данные) переписываются из дополнительной памяти на магнитную ленту для дальнейшего применения.

Программное обеспечение работы с магнитофоном значительно увеличивает достоверность процесса записи-считывания. При записи на магнитную ленту файлы разбиваются на блоки объемом 1 К. Каждый блок снабжается заголовком со следующей информацией: длина (2 байта), номер (2 байта) блока; имя (8 байт) и тип (3 байта) файла; контрольная сумма (2 байта).

При записи файла на ленту создаются две одинаковые копии. В ходе операции чтения программное обеспечение проверяет контрольную сумму для каждого блока файла. В случае несовпадения контрольных сумм (подсчитанной и записанной в заголовке блока),

Таблица 1

Подпрограмма «Чтение сектора»

Назначение —
Пересылка 128 байт информации из дополнительной памяти в основную

Входные параметры —
(BUFAD) — адрес буфера в основной памяти,
(DSKN) — номер «диска»

Использование регистров —
A, B, D, E, H, L, SP

Обращение к подпрограммам —
COMAD — определение физического адреса «сектора» в дополнительной памяти

Результат работы —
по возврату из подпрограммы в основной памяти заполнен буфер размером 128 байт

```
ONSP   EQU   40H      ; ПОРТ БЛОКА УПРАВ-
                      ЛЕНИЯ
READ:  CALL  COMAD    ; ОПРЕДЕЛЕНИЕ АДРЕ-
                      СА СЕКТОРА
       LXI   H, 0     ;
       DAD   SP       ; HL=SP
       SHLD  OLDSP    ; ЗАПОМНИТЬ SP В ПА-
                      МЯТИ
       LHLD  PHAD     ; HL = АДРЕСУ «СЕКТО-
                      РА»
       SPHL           ; SP = АДРЕСУ «СЕКТО-
                      РА»
       LHLD  BUFAD    ; HL = АДРЕСУ БУФЕРА
       MVI   B, 64D   ; B — СЧЕТЧИК ПАР БАЙ-
                      ТОВ
       LDA   DSKN     ; A = НОМЕРУ «ДИСКА»
       OUT   ONSP     ; РАЗРЕШИТЬ ОБРАЩЕ-
                      НИЕ К «ДИСКУ»
RDSK:  POP   D        ; ПЕРЕСЛАТЬ В DE ОЧЕ-
                      РЕДНЫЕ
                      ; ДВА БАЙТА
       MOV   M, E     ; ПЕРЕСЛАТЬ 1-Й БАЙТ
                      В БУФЕР
       INX   H
       MOV   M, D     ; ПЕРЕСЛАТЬ 2-Й БАЙТ
                      В БУФЕР
       INX   H
       DCR   B        ; СЧЕТЧИК = СЧЕТ-
                      ЧИК — 1
       JNZ   RDSK     ; СЧЕТЧИК=0? НЕТ —>
                      ПОВТОР
ENDIO: MVI   A, 0FFH  ; ЗАВЕРШЕНИЕ ОБРА-
                      ЩЕНИЯ К «ДИСКУ»
       OUT   ONSP     ; ЗАПРЕТИТЬ ОБРАЩЕ-
                      НИЕ К «ДИСКУ»
       LHLD  OLDSP    ; HL = ПРЕЖНЕМУ ЗНА-
                      ЧЕНИЮ SP
       SPHL           ; SP = ПРЕЖНЕМУ ЗНА-
                      ЧЕНИЮ
       RET
```

данный блок помечается как дефектный. После прочтения всей первой копии файла из второй копии считываются только дефектные блоки. «Собранный» таким образом файл записывается в дополнительную память — «квазидиск». Для ввода и вывода байта информации данная программа обращается к программе монитор. При наличии магнитофона с электронным управлением режимами работы лентопротяжного механизма возможна работа с автоматическим поиском файлов на ленте.

Таблица 2

Подпрограмма «Запись сектора»

Назначение —
пересылка 128 байт из основной памяти в дополнительную

Входные параметры —
(BUFAD) — адрес буфера в основной памяти,
(DSKN) — номер «диска»

Использование регистров —
A B, D, E, H, L, SP

Обращение к подпрограммам —
COMAD — вычисление физического адреса «сектора» в дополнительной памяти

Результат работы —
по возврату из подпрограммы 128 байт из основной памяти записаны на «диск»

```
WRITE: CALL   COMAD   ; ОПРЕДЕЛЕНИЕ АДРЕСА «СЕКТОРА»
       LXI    D, 128D ; DE = 128 (1 «СЕКТОР»)
       DAD    D       ; HL = АДРЕСУ ПОСЛЕДНЕГО БАЙТА «СЕКТОРА» + 1
       SHLD   PHAD    ; ЗАПОМНИТЬ ЕГО В ПАМЯТИ
       LXI    H, O
       DAD    SP      ; HL = SP
       SHLD   OLDSP:  ; ЗАПОМНИТЬ SP В ПАМЯТИ
       LHLD   PHAD    ;
       SPHL           ; SP = АДРЕСУ ПОСЛЕДНЕГО БАЙТА
                      ; В «СЕКТОРЕ» + 1
       LHLD   BUFAD:  ; HL = АДРЕСУ БУФЕРА
       DAD    D       ; HL = АДРЕСУ БУФЕРА + 128
       DCX    H       ; HL = АДРЕСУ ПОСЛЕДНЕГО БАЙТА
                      ; В БУФЕРЕ
       MVI    B, 64D  ; B — СЧЕТЧИК ПАР БАЙТОВ
       LDA    DSKN    ; A = НОМЕРУ «ДИСКА»
       OUT    ONSP    ; РАЗРЕШИТЬ ОБРАЩЕНИЕ К «ДИСКУ»
WDSK:  MOV    D, M    ; СТАРШИЙ БАЙТ ИЗ БУФЕРА В D
       DCX    H
       MOV    E, M    ; МЛАДШИЙ БАЙТ ИЗ БУФЕРА В E
       DCX    H
       PUSH   D       ; ЗАПОМНИТЬ ИХ НА «ДИСКЕ»
       DCR    B       ; СЧЕТЧИК = СЧЕТЧИК — 1
       JNZ    WDSK    ; СЧЕТЧИК = 0? НЕТ —> ПОВТОР
       JMP    ENDI0   ; ЗАВЕРШЕНИЕ ПРОЦЕДУРЫ ЗАПИСИ
; ЗАРЕЗЕРВИРОВАННЫЕ ЯЧЕЙКИ ОЗУ ДЛЯ ПОДПРОГРАММ ЧТЕНИЯ
; И ЗАПИСИ
OLDSP  DS     2       ; ПРЕЖНЕЕ ЗНАЧЕНИЕ SP
DSKN:  DS     1       ; НОМЕР БЛОКА ДОПОЛНИТЕЛЬНОЙ
                      ; ПАМЯТИ («ДИСКА»)
BUFAD: DS     2       ; АДРЕС БУФЕРА В ОСНОВНОЙ ПАМЯТИ
PHAD:  DS     2       ; АДРЕС «СЕКТОРА» В ДОПОЛНИТЕЛЬНОЙ
                      ; ПАМЯТИ
```

МикроЭВМ работает под управлением операционной системы, совместимой с CP/M80. При этом функционируют трансляторы с языков программирования ассемблер, Паскаль, Бейсик; экранный редактор текста; отладчик и другие программы.

Такие микроЭВМ могут быть объединены в локальную сеть в пределах здания 4-проводной физической линией связи. Программное обеспечение ориентировано на структуру сети типа «звезда» (в качестве центральной ЭВМ применима СМ-1800 на базе БИС КР580ИК80А [3]). Сетевое программное обеспечение совместимо с программным обеспечением MP/M—CP/NET.

Разработанный вариант микроЭВМ конструктивно выполнен в корпусе размером 135×105×400 мм и может служить в качестве бытовой или школьной ЭВМ.

ЛИТЕРАТУРА

Зеленко Г. В., Панов В. В., Попов С. Н. Радиолюбителю о микропроцессорах и микроЭВМ. — Радио, 1982, № 9—12; 1983, № 2—4, 6—12.

Спарлинг Б. Дж. Периферийная память для микропроцессора Z80 на динамических ЗУПВ емкостью 64 К. — Электроника, 1983, № 5, с. 77.

Прохоров Н. Л., Ландау И. Я. МикроЭВМ СМ-1800 и ее программное обеспечение. — Микропроцессорные средства и системы, 1984, № 2, с. 28.

Статья поступила 24 апреля 1984 г.

УДК 681.3.—181.1.06

В. Г. Майоров, А. И. Гаврилов

МОНИТОР ОЦЕНОЧНОГО МОДУЛЯ ДЛЯ МИКРОПРОЦЕССОРА КР580ИК80

При разработке целевых программ для микропроцессорных систем (МПС) необходимы специальные вспомогательные программы. Управляющая программа-монитор выполняет простейшие функции системного программного обеспечения. Ее основное назначение — отладка программ в машинных кодах.

Предлагается простой и эффективный вариант монитора для оценочного модуля. Достоинства монитора: минимальный необходимый объем ПЗУ системы (254 байта); наличие набора стандартных подпрограмм, которые могут быть эффективно использованы оператором при отладке программ; возможность работы с простейшей клавиатурой и индикацией.

Для функционирования монитора требуется обращение к ряду устройств микропроцессорной системы. Структурная схема оценочного модуля приведена на рис. 1. Адреса, присвоенные устройствам, приведены в табл. 1. Монитор работает с клавиатурой, электрическая

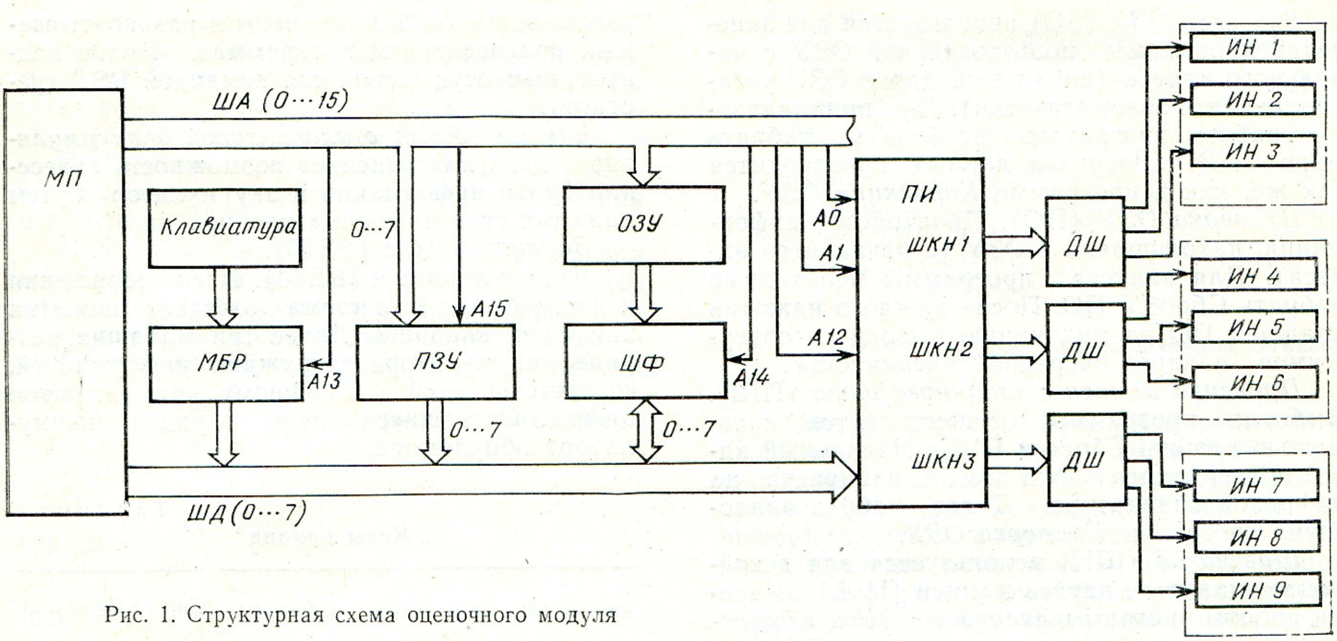


Рис. 1. Структурная схема оценочного модуля

принципиальная схема которой изображена на рис. 2. Индикация вводимой и выводимой информации осуществляется с помощью параллельного интерфейса КР580ВВ55, схемы дешифрации и трех триад ламп.

Таблица 1

Адреса устройств

Устройство	Адрес восьмеричный	
	начальный	конечный
Клавиатура	240 000	—
Индикация	220 000	220 003
ПЗУ	000 000	000 377
ОЗУ	300 000	303 377
Стек (входит в состав ОЗУ)	303 300	303 377

В данной программе триада ламп, соответствующая каналу ШКН1 (КР580ВВ55), служит для индикации байта данных; ШКН2 — для старшего байта адреса, ШКН3 — для младшего байта адреса. Монитор занимает в ПЗУ системы 254 байта (например, для кристалла КР556РТ4). Набор программ, входящих в монитор, можно разделить на три основные части: функциональные, тестирующие и стандартные подпрограммы.

Программа монитора приведена в табл. 2.

Функциональные программы. *Коррекция ОЗУ* (КО). С её помощью можно исправить неверно записанную информацию в произвольной ячейке ОЗУ и загрузить программы пользователя в любую область ОЗУ. Выборка программы осуществляется нажатием клавиш Сброс и КО, затем на цифровой клавиатуре шестью нажатиями клавиш набирается восьмеричный адрес нужной ячейки ОЗУ, начиная со старшего байта. Набранный код адреса высвечивается на индикаторе в адресных триадах. Далее производится загрузка информации: одному байту соответствует нажатие трех клавиш. Данные вводятся в восьмеричном коде. С помощью индикации можно проследить за формированием очередного байта вводимых данных и за изменением адреса. Адреса вводятся автоматически.

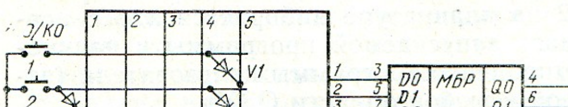


Рис. 2. Схема электрическая принципиальная клавиатуры

Загрузка ОЗУ (ЗО) используется для занесения программы пользователя в ОЗУ с начального адреса (начальный адрес ОЗУ устанавливается автоматически). Для инициализации работы программы необходимо набрать Сброс и ЗО. Загрузка данных производится так же, как в программе *Коррекция ОЗУ*.

Проверка ОЗУ (ПО). Проверяется информация, находящаяся в ОЗУ (с начального адреса). Для запуска программы необходимо набрать Сброс и ПО. После каждого нажатия клавиши ПО на индикацию выводится содержимое и адрес очередной ячейки ОЗУ.

Проверка памяти с заданного шага (ПШ). Выборка программы осуществляется с помощью клавиш Сброс и ПШ. Начальный адрес проверяемого блока памяти набирается на цифровой клавиатуре. Далее работа аналогична программе *Проверка ОЗУ*.

Шаг назад (ШН) используется для декремента указателя адреса памяти (H, L) во время работы программ *Загрузка ОЗУ и Коррекция ОЗУ*. После нажатия клавиши ШН на индикацию выводится: 000 в триаде данных и уменьшенный на единицу текущий адрес ячейки ОЗУ. Затем вводится байт данных, и загрузка памяти продолжается в обычном порядке.

Пуск 1. После нажатия клавиш Сброс и Пуск 1 система выполняет программу пользователя, записанную с начального адреса. ОЗУ устанавливается автоматически).

Пуск 2. После нажатия клавиш Сброс и Пуск 2 на клавиатуре набирается адрес первого шага запускаемой программы и начинается выполнение программы пользователя (записанной в любой области ОЗУ).

Тестирующие программы. *Тест индикации* служит для проверки работоспособности устройств системы, связанных с выводом данных на индикацию. Для пуска программы нажимаются клавиши Сброс и ТИ. На всех триадах индикации последовательно высвечиваются следующие комбинации цифр: 000, 111, 222, 333, 044, 155, 266, 377.

Тест ОЗУ (ТО). Выборка программы производится клавишами Сброс и ТО. Если все ячейки ОЗУ исправны, то на индикацию выводится следующая комбинация цифр: 377 304 000. Если же в цикле запись-считывание для данной ячейки происходит искажение информации, то содержимое и адрес неисправной ячейки выводятся на индикацию. Если перевести полученный байт данных в двоичный код, то ноль среди единиц или единица среди нулей указывают на неисправный разряд. Продолжить тест после обнаружения ошибки можно с помощью нажатия клавиши ТО.

Стандартные подпрограммы. Используются в управляющих и тестирующих программах, также могут быть применены в разрабатываемых пользователем программах. Вызов подпрограмм осуществляется командой RST (рестарт).

Чтение байта с клавиатурой в аккумулятор (RST2). Определяет возможность занесения байта информации в акуумулятор путем нажатия трех цифровых клавиш.

Задержка 0,6 с (RST3).

Чтение клавиши (RST4). После обращения к подпрограмме система ожидает нажатия цифровой клавиши. Далее три младших разряда аккумулятора загружают информацией, соответствующей двоичному коду нажатой цифры. Оставшиеся пять разрядов аккумулятора обнуляются.

Таблица 2

Коды команд

00000	3E	80	32	03	90	31	FF	C3
00000	E7	C6	EB	26	00	6F	6E	E9
00016	C3	33	00	E6	0F	C3	1B	00
00024	C3	5F	00	FE	08	C3	F8	00
00032	DF	C3	6F	00	3B	C3	9B	00
00040	C3	7A	00	2B	3B	C3	24	00
00048	C3	88	00	D5	C5	16	00	06
00056	03	00	E7	05	48	CA	57	00
00064	0D	CA	49	00	1E	06	C3	4B
00072	00	1E	03	07	1D	C2	4B	00
00080	B2	57	32	00	90	C3	39	00
00088	B2	32	00	90	C1	D1	C9	E5
00096	F5	21	50	46	3E	00	2B	B4
00104	B5	C2	64	00	F1	E1	C9	3A
00112	00	A0	C6	20	CA	6F	00	C3
00120	13	00	F5	32	00	90	7C	32
00128	01	90	7D	32	02	90	F1	C9
00136	F5	D7	57	D7	5F	F1	C9	F7
00144	EB	C3	9B	00	EB	C3	8F	00
00152	21	00	C0	3E	00	EF	D7	77
00160	DF	23	C3	9B	00	21	00	C0
00168	E7	7E	EF	23	C3	A8	00	16
00176	48	3E	00	67	6F	EF	C6	49
00184	DF	BA	C2	B3	00	C7	21	00
00192	C0	3E	00	77	7E	C6	00	C2
00200	D9	00	3E	FF	77	7E	3C	C2
00208	D9	00	23	7C	E6	04	CA	C1
00216	00	7E	EF	E7	C3	C1	00	D7
00224	67	D7	6F	3E	00	EF	DF	E9
00232	C3	00	C0	8F	98	A5	AF	BE
00240	F3	E8	DF	F7	EB	C3	A8	00
00248	CA	2B	00	E6	07	C9	00	00

Вывод на индикацию (RST5). Содержимое регистров выводится на индикацию. Подпрограмму удобно использовать в случае, если регистровая пара (H, L) микропроцессора служит указателем адреса ячеек памяти.

Все перечисленные подпрограммы сохраняют содержимое регистров общего назначения неизменным.

При применении предложенного монитора в микропроцессорной системе исключаются устройства, работающие в режиме прерывания процессора, поскольку при вызове стандартных подпрограмм используются команды RST.

Статья поступила 17 июля 1984 г.

УДК 681.3

Е. Н. Федонов

МИКРОКОНТРОЛЛЕРНОЕ УСТРОЙСТВО ПЕРВИЧНОЙ ОБРАБОТКИ ИНФОРМАЦИИ

Микроконтроллерное устройство можно использовать для систем учета, контроля, охраны на железнодорожном и автомобильном транспорте, в АСУ технологическими процессами для индивидуального отбора объектов.

Включение в систему нескольких микроконтроллерных устройств (МКУ) первичной обработки информации (распределенная обработка) освобождает центральную управляющую ЭВМ от рутинных операций, а применение микропроцессорного комплекта позволяет для разнофункциональных МКУ использовать аналогичные по архитектуре устройства, отличающиеся в основном только программным обеспечением. В этом случае задачи управления решаются эффективнее.

Микроконтроллерное устройство на базе микропроцессорного комплекта КР580 достоверно (на фоне помех) принимает цифровой код, поступающий в последовательном виде от приемного устройства распознавания. Затем преобразует его в 15-разрядный код и в код 7-сегментных индикаторов, передает цифровой код в центральную ЭВМ комплекса, выдает десятичный номер на цифровой дисплей, запрещает повторный вывод одного и того же кода в центральную ЭВМ.

Структура входных сигналов МКУ представлена на рис. 1. Между 1-м и 17-м тактовыми импульсами находится информационное поле, а между 17-м и 1-м (следующего цикла) тактами — синхропауза (длительностью

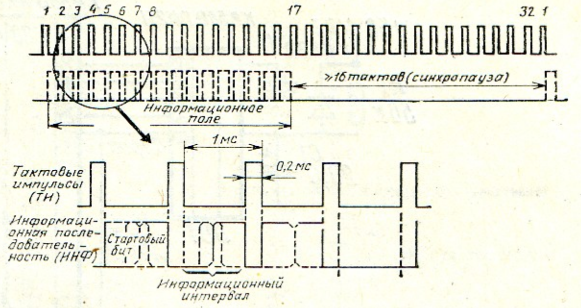


Рис. 1. Структура входных информационных сигналов

не менее 16 тактов). Информационные биты передаются кодом с возвратом к нулю; значению бита, равному 1, соответствует высокий уровень, значению 0 — низкий. Первый бит информационного поля воспринимается МКУ как стартовый, остальные — как информационные. Информационные биты расположены в промежутках между тактовыми импульсами (информационный интервал). Информация может быть получена в любой момент информационного интервала. Веса информационных битов расположены в возрастающем порядке слева направо. Данная структура передаваемой информации позволяет кодировать до 32768 комбинаций при 15-разрядном коде. В описываемом МКУ выдается только 11-разрядный код, что соответствует 2048 комбинациям.

Достоверность принимаемой информации повышается многократным приемом с анализом на совпадение. Кроме того, запрещен прием нулевого кода. Это защищает систему от приема единичных помех в момент отсутствия информации. Количество совпадений можно выбрать в соответствии с условиями приема, но в этом случае время обработки увеличивается до $N \cdot 32$ мкс, где N — число совпадений, а 32 мкс — период следования информационных полей.

Принципиальная схема МКУ (рис. 2) построена в основном на БИС микропроцессорного комплекта КР580 и включает в себя:

Центральный процессорный элемент	(ЦПЭ); DD2	КР580ИК80
Задающий генератор	(Г); DD1	КР580ГФ24
Схему формирования выборок	(СФВ); DD4, DD5	К155ЛН6, К155, ЛА3
Оперативное запоминающее устройство	(ОЗУ); DD10, DD11	К541РУ2
Постоянное запоминающее устройство	(ПЗУ); DD12	К573РФ12

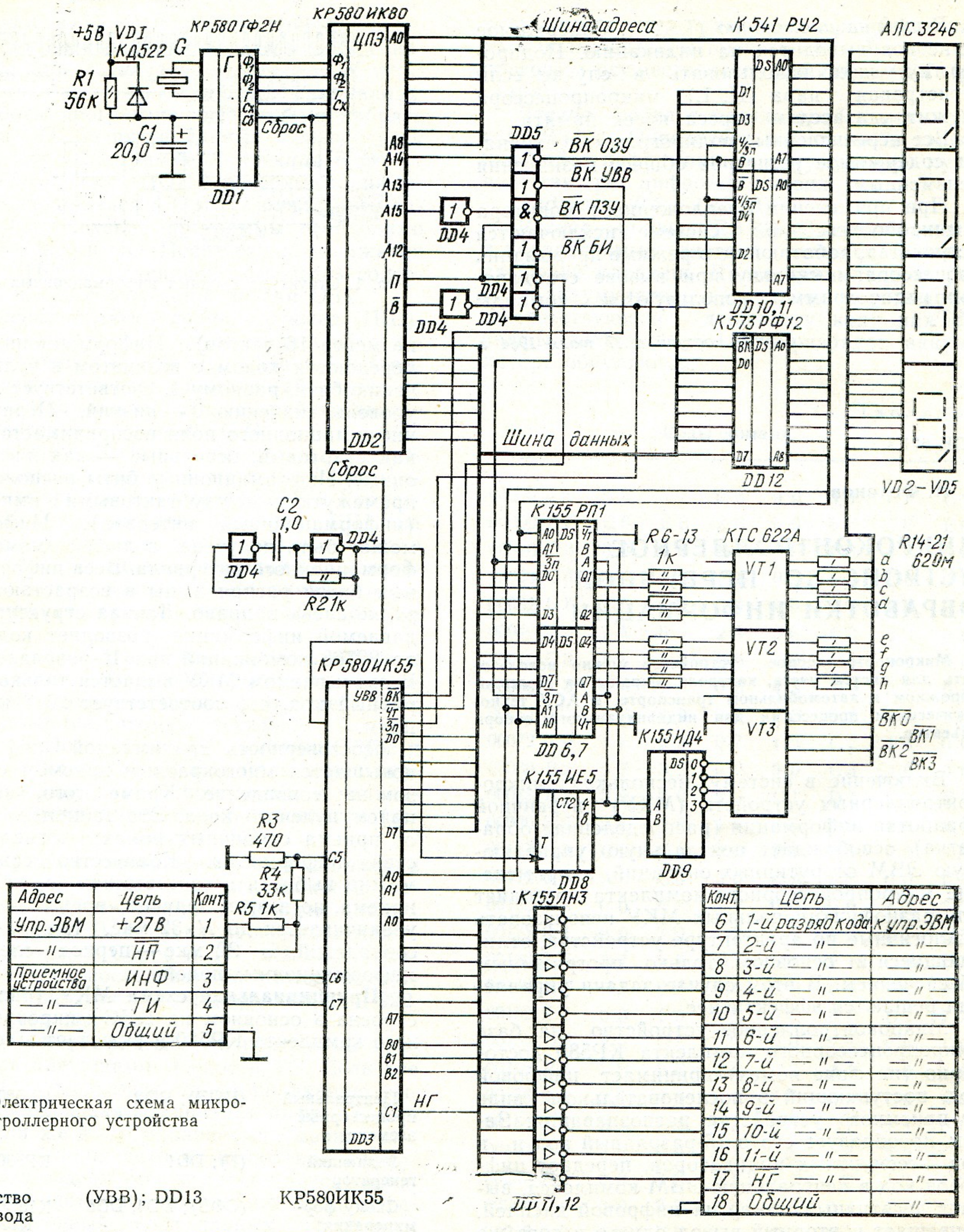


Рис. 2. Электрическая схема микроконтроллерного устройства

Устройство ввода-вывода	(УВВ); DD13	КР580ИК55
Регистр индикации	(РИ); DD6, DD7	К155РП1 (см. рис. 2)
Устройство управления динамической индикацией	(УДИ); DD4, C2, R2, DD8, DD9	К155ИЕ5, К155, ИД4
Блок индикаторов	(БИ); VT1—VT3, VD2—VD5	АЛС324Б АЛС324Б
Выходной формирователь	(ВФ); DD11, DD12	К155ЛН3

```
                              ;ОСНОВНАЯ ПРОГРАММА РАБОТЫ МКУ РАСПОЗНАВАНИЯ
                              ;НОМЕРА ЖИВОТНОГО

 10                                    RADIX   16
 C010                         СТЕК     EQU     0C010    ;АДРЕС УКАЗАТЕЛЯ
                                                        ;СТЕКА
 9003                         ДИСПЛ    EQU     9003     ;АДРЕС СТАРШЕГО РАЗ-
                                                        ;РЯДА ДИСПЛЕЯ
 00A3                         РУС      EQU     0A3      ;АДРЕС РЕГИСТРА УПР.
                                                        ;СЛОВА К580 ИК55
 A0A0                         ПОРТ.А   EQU     0A0A0    ;АДРЕС ДВОЙНОГО ПОР-
                                                        ;ТА АВ
 00A2                         ПОРТ.С   EQU     0A2      ;АДРЕС ПОРТА С

 0000                                  ORG     0

 0000  31 10 C0                        LXI     SP,СТЕК
 0003  01 00 00                        LXI     B,0
 0006  C5                              PUSH    B

 0007  3E 88                           MVI     A,88     ;ПРОГРАММИРОВАНИЕ
                                                        ;К580 ИК55
 0009  D3 A3                           OUT     РУС      ;ПОРТ.А,ПОРТ.В,ПОРТ.
                                                        ;С(0-3) НА ВЫВОД
                                                        ;ПОРТ.С(4-7) НА ВВОД

 000B  06 03                  СТАРТ:   MVI     B,3      ;ОПРЕДЕЛИТЬ КОЛИЧЕСТ-
                                                        ;ВО СРАВНЕНИЙ КОДА
 000D  EB                              XCHG             ;В DE ХРАНИТСЯ ПРЕДЫ-
                                                        ;ДУЩИЙ КОД
 000E  AF                     ОСП:     XRA     A        ;ОБНУЛИТЬ СЧЕТЧИК КО-
                                                        ;ЛИЧЕСТВА ПАУЗ
 000F  4F                              MOV     C,A

 0010  DB A2                  ВВИНФ:   IN      ПОРТ.С   ;ВВОД ИНФОРМАЦИИ И
                                                        ;ТАКТОВ (КАН. С6,С7)
 0012  E6 C0                           ANI     0C0      ;АНАЛИЗ НАЛИЧИЯ ТАКТА
 0014  17                              RAL
 0015  DA 10 00                        JC      ВВИНФ

                                   ПОИСК СИНХРОПАУЗЫ
 0018  CD B8 00                        CALL    АНИ      ;АНАЛИЗ НАЛИЧИЯ ИНФОР-
                                                        ;МАЦИИ НА ВСЕМ ИНФОРМ.
                                                        ;ПОЛЕ
 001B  FE 00                           CPI     0
 001D  C2 0E 00                        JNZ     ОСП
 0020  0C                              INR     C
 0021  79                              MOV     A,C
 0022  FE 10                           CPI     16D
 0024  C2 10 00                        JNZ     ВВИНФ    ;ПЕРЕХОД, ЕСЛИ КОЛИ-
                                                        ;ЧЕСТВО ПРИНЯТЫХ
                                                        ;ПОДРЯД ПАУЗ < 16

                              ;    ПОИСК СИНХРОИМПУЛЬСА
 0027  DB A2                  ОСИ:     IN      ПОРТ.С
 0029  E6 40                           ANI     40
 002B  CA 27 00                        JZ      ОСИ

                              ;ПРЕОБРАЗОВАНИЕ В Н И ПЕРЕСЫЛКА В L 8-МИ
                              ;МЛАДШИХ РАЗРЯДОВ КОДА
 002E  0E 01                           MVI     C,1      ;СЧЕТЧИК НА 8 ЦИКЛОВ
 0030  CD 9B 00                        CALL    V
 0033  7C                              MOV     A,H
 0034  07                              RLC
 0035  6F                              MOV     L,A
                              ;ПРЕОБРАЗОВАНИЕ В Н 7-МИ СТАРШИХ РАЗР. КОДА
 0036  0E 02                           MVI     C,2      ;СЧЕТЧИК НА 7 ЦИКЛОВ
 0038  CD 9B 00                        CALL    V

 003B  CD C6 00                        CALL    РАВНО    ;СРАВНИТЬ DE И HL,
                                                        ;ПЕРЕХОД ПРИ НЕСОВ-
                                                        ;ПАДЕНИИ
 003E  C2 0B 00                        JNZ     СТАРТ
 0041  05                              DCR     B        ;АНАЛИЗ КОЛ-ВА СОВ-
                                                        ;ПАДЕНИЙ, ПЕРЕХОД,
                                                        ;ЕСЛИ В>0
 0042  C2 10 00                        JNZ     ВВИНФ

                              ;ЗАЩИТА ОТ ЕДИНИЧНЫХ ПОМЕХ
 0045  7C                              MOV     A,H
 0046  B3                              ORA     E
 0047  CA 0B 00                        JZ      СТАРТ

 004A  E5                              PUSH    H

 004B  CD CC 00                        CALL    ДЕСКОР   ;ДЕСЯТИЧНАЯ КОРРЕК-
                                                        ;ЦИЯ СОДЕРЖИМОГО HL,
 004E  4F                              MOV     C,A      ;РЕЗУЛЬТАТ В А И С
 004F  CD CC 00                        CALL    ДЕСКОР

                              ;ИНДИКАЦИЯ КОДА НА ДИСПЛЕЕ
 0052  11 03 90                        LXI     D,ДИСПЛ
 0055  CD 83 00                        CALL    ИНДИ
 0058  79                              MOV     A,C
 0059  CD 83 00                        CALL    ИНДИ

 005C  E1                              POP     H
 005D  D1                              POP     D
 005E  CD C6 00                        CALL    РАВНО    ;СРАВНИТЬ DE И HL,
 0061  E5                              PUSH    H        ;НОВЫЙ КОД В СТЕК
 0062  CA 0B 00                        JZ      СТАРТ    ;ПЕРЕХОД ПРИ РАВЕН-
                                                        ;СТВЕ КОДОВ

                              ;ВЫВОД КОДА В МАГИСТРАЛЬ
 0065  22 A0 A0                        SHLD    ПОРТ.А
 0068  21 10 27                        LXI     H,2710   ;ЗАГРУЗИТЬ КОНСТАНТУ
                                                        ;НА 0,3 СЕК.
 006B  3E 03                           MVI     A,3      ;ВЫВОД СИГН. "НГ" В
 006D  D3 A3                           OUT     РУС      ;КАН.С1
 006F  DB A2                  X:       IN      ПОРТ.С   ;АНАЛИЗ СИГН. "НП"
                                                        ;(КАН.С5)
 0071  E6 20                           ANI     20
 0073  C2 7C 00                        JNZ     СБРОС
 0076  2B                              DCX     H
 0077  7D                              MOV     A,L
 0078  B4                              ORA     H
 0079  C2 6F 00                        JNZ     X        ;ОЖИДАНИЕ ПРИХОДА "НП"
                                                        ;0,3 СЕК.

 007C  3E 02                  СБРОС:   MVI     A,2      ;СБРОС СИГН. "НГ"
 007E  D3 A3                           OUT     РУС
 0080  C3 0B 00                        JMP     СТАРТ

                              ;П/П ИНДИКАЦИИ
 0083  CD 86 00               ИНДИ:    CALL    ИН
 0086  21 DB 00               ИН:      LXI     H,ТАБЛ.1
 0089  07                              RLC
 008A  07                              RLC
 008B  07                              RLC
 008C  07                              RLC
 008D  47                              MOV     B,A
 008E  E6 0F                           ANI     0F
 0090  85                              ADD     L
 0091  D2 95 00                        JNC     K
 0094  24                              INR     H
 0095  6F                     K:       MOV     L,A
 0096  7E                              MOV     A,M
 0097  12                              STAX    D
 0098  1B                              DCX     D
 0099  78                              MOV     A,B
 009A  C9                              RET

                              ;ПРЕОБРАЗОВАНИЕ ПОСЛЕДОВАТЕЛЬНОЙ ИНФОРМАЦИИ
                              ;В ПАРАЛЛЕЛЬНУЮ
 009B  26 00                  V:       MVI     H,0
 009D  DB A2                  J:       IN      ПОРТ.С
 009F  E6 80                           ANI     80
 00A1  CA 9D 00                        JZ      J        ;ПЕРЕХОД ПРИ ОТСУТ-
                                                        ;СТВИИ ТАКТА
 00A4  DB A2                  Z:       IN      ПОРТ.С
 00A6  E6 C0                           ANI     0C0
 00A8  17                              RAL
 00A9  DA A4 00                        JC      Z        ;ПЕРЕХОД ПРИ НАЛИЧИИ
                                                        ;ТАКТА
 00AC  CD BA 00                        CALL    W        ;СЧИТАТЬ ИНФ. ИНТЕРВАЛ
 00AF  0F                              RRC              ;В А
 00B0  67                              MOV     H,A
 00B1  79                              MOV     A,C
 00B2  17                              RAL
 00B3  4F                              MOV     C,A
 00B4  D8                              RC               ;ВЫХОД ПО ПЕРЕНОСУ
                                                        ;(8 ИЛИ 7 ЦИКЛОВ)
 00B5  C3 A4 00                        JMP     Z        ;ПРОДОЛЖИТЬ ОЧЕРЕДНОЙ
                                                        ;ЦИКЛ

                              ;ОПРЕДЕЛЕНИЕ НАЛИЧИЯ ИНФОРМАЦИИ НА ИНФ. ИНТЕР-
                              ;ВАЛЕ
 00B8  26 00                  АНИ:     MVI     H,0
 00BA  B4                     W:       ORA     H        ;ПОЛУЧЕНИЕ В СТАРШЕМ
                                                        ;РАЗРЯДЕ РЕЗУЛЬТИРУЮ-
                                                        ;ЩЕГО ЗНАЧЕНИЯ ЗА ВЕСЬ
 00BB  67                              MOV     H,A      ;ИНФ. ИНТЕРВАЛ
 00BC  DB A2                           IN      ПОРТ.С
 00BE  E6 C0                           ANI     0C0
 00C0  17                              RAL
 00C1  D2 BA 00                        JNC     W
 00C4  7C                              MOV     A,H
 00C5  C9                              RET


                              ;СРАВНЕНИЕ СОДЕРЖИМОГО HL И DE
 00C6  7B                     РАВНО:   MOV     A,E
 00C7  BD                              CMP     L
 00C8  C0                              RNZ              ;ВЫХОД ЕСЛИ НЕ РАВНО
 00C9  57                              MOV     D,A
 00CA  BC                              CMP     H        ;АНАЛИЗ ФЛАЖКА В ОС-
 00CB  C9                              RET              ;НОВНОЙ ПРОГРАММЕ

                              ;ДЕСЯТИЧНАЯ КОРРЕКЦИЯ
 00CC  1E 11                  ДЕСКОР:  MVI     E,17D
 00CE  AF                              XRA     A
 00CF  1D                     Ж:       DCR     E
 00D0  C8                              RZ
 00D1  29                              DAD     H
 00D2  8F                              ADC     A
 00D3  27                              DAA
 00D4  D2 CF 00                        JNC     Ж
 00D7  23                              INX     H
 00D8  C3 CF 00                        JMP     Ж

                              ;ТАБЛИЦА ШИФРАЦИИ КОДОВ
 00DB  3F 06 5B               ТАБЛ.1:  DB      3F,06,5B,4F,66,6DH,7DH,07,7F,6F
 00DE  4F 66 6D
 00E1  7D 07 7F
 00E4  6F
                                       END
```

Рис. 3. Программа работы МКУ

ЦПЭ управляет работой всего устройства по адресной шине и шине данных с помощью сигналов управления направлением движения информации ($\overline{В}$, ПР) по программе, записанной в ПЗУ.

Задающий генератор осуществляет тактирование ЦПЭ: вырабатывает два сигнала $Ф_1$ и $Ф_2$ с помощью цепи VD1—R1—C1, обеспечивает начальную установку ЦПЭ в момент подачи напряжения питания. Применение кварцевого резонатора G в схеме Г необходимо для стабилизации временны́х характеристик системы. В простых МКУ можно ограничиться емкостью 5...15 пФ.

СФВ осуществляет адресацию всех устройств МКУ путем выделения соответствующей комбинации на адресной шине. Устройствам присвоены следующие адреса обращения:

ПЗУ	0000—ОЕЕЕ	(Адреса даны в гексадецимальном коде)
ОЗУ	С000—СЕЕЕ	

РИ	9000—9003	(Остальные комбинации не используются)
УВВ	А0, 01, 02, А3	(Соответственно по портам А, В, С. РУС).

Использование только двух сигналов для управления передачей информации ($\overline{В}$, $\overline{ПР}$) исключает необходимость организации системного контроллера и позволяет обращаться ко всем устройствам, подключенным к шине данных, как к однотипным (ячейки памяти или УВВ), т. е. и командами работы с памятью, и командами работы с УВВ. Это создает более свободные условия для программиста. Так, например, можно одной командой обратиться сразу к двум портам УВВ.

УВВ используется для: ввода информационной и тактовой последовательностей из приемного устройства распознавания через каналы С6 и С7 в МКУ; вывода через каналы А0-В2 двоичного кода и сигнала высокого уровня «Номер готов» (НГ) по каналу С1 в шину ЭВМ через ВФ; считывания сигнала низкого уровня «Номер принят» ($\overline{НП}$) с шины ЭВМ через канал С5. Комбинация режимов УВВ позволяет расширить функциональные возможности МКУ.

В *ОЗУ* организуется стек и хранятся промежуточные результаты. В схеме предусмотрен значительный запас по емкости ОЗУ (1 Кбайт), что позволяет (так же, как и УВВ), не изменяя архитектуры системы, расширить ее функциональные возможности.

РИ предназначен для записи кода индикации, поступающего с шины данных. В РИ записывается код 7-сегментных индикаторов, получение которого из двоично-десятичной формы осуществляется МКУ с помощью табличного способа. Считывается код из РИ и передается в блок 7-сегментных индикаторов по сигналам УДИ.

УДИ обеспечивает подекадную выборку двоично-десятичного кода из РИ и формирование сигналов последовательного переключения индикаторов БИ с частотой $F=1/R_2C_2$.

БИ представляет собой дисплей, состоящий из четырех 7-сегментных светодиодных индикаторов и ключей управления, осуществляющих коммутацию анодных токов сегментов индикаторов и токов общих катодов.

Рассмотренное МКУ можно использовать в информационно-измерительной системе автоматизированного кормления животных с учетом их индивидуальных физиологических особенностей, разработанной ВИЭСХ для животноводческих комплексов.

Для распознавания животных каждому из них укрепляется на ошейнике (или вшивается под кожу) датчик-приемник. Номер животного записывается в виде кодовой группы. При прохождении зоны распознавания животное с датчиком оказывается между антеннами радиопередатчика импульсов с высокочастотным заполнением. Датчик возбуждается и генерирует ответную серию высокочастотных импульсов (в соответствии с кодом датчика), принимаемую антенной передатчика, переключающегося в режим приема. Прием импульсов на фоне помех, их обработка и обеспечивается МКУ. Применение МКУ позволило существенно упростить электрическую схему устройства распознавания, добиться большей логической гибкости и адаптивности устройства.

Данное МКУ может служить базовым для создания устройства сбора и обработки первичной информации в АСУ и других подобных системах. На основе этого МКУ созданы устройства взвешивания животных и контроля процесса сушки, причем изменения архитектуры коснулись только периферийных устройств, хотя программное обеспечение было разработано заново. По расчетам специалистов ВИЭСХ, внедрение МКУ на 25% из существующих 2067 (на 01.01.82 г.) молочных комплексах со средним поголовьем 650 коров позволит получить годовой экономический эффект 800 тыс. руб. [1]*. На рис. 3 приведена программа работы МКУ.

---

* Краусп В. Р. Комплексная автоматизация в промышленном животноводстве. — М.: Машиностроение, 1980, с. 205.

Статья поступила 11 мая 1984 г.

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Б. М. Малашевич, В. А. Шахнов, Э. И. Коночкин

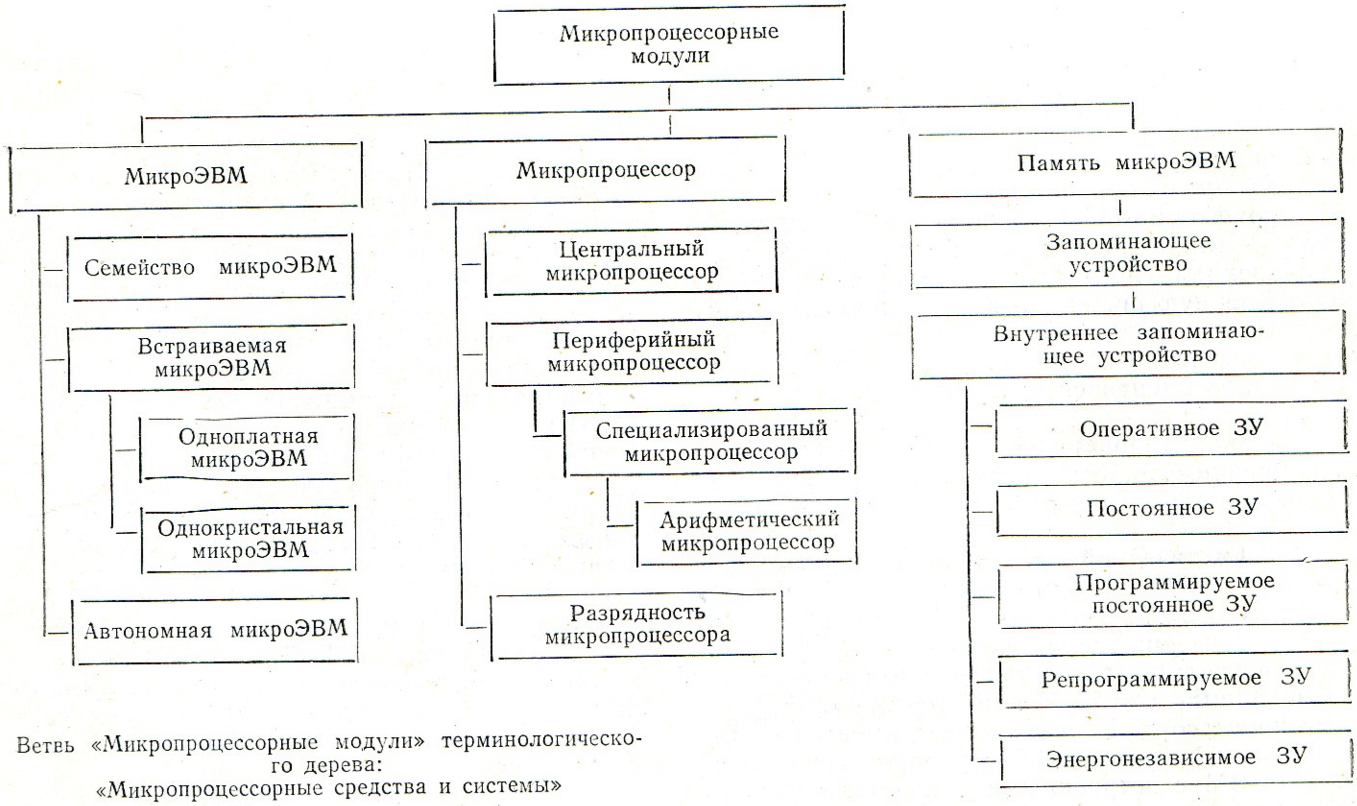
МИКРОПРОЦЕССОРНЫЕ МОДУЛИ

Во второй статье из серии «Термины и определения»* рассматривается часть ветви терминологического дерева «микропроцессорные модули» (см. рисунок), охватывающая основные системообразующие модули: микроЭВМ, микропроцессоры и внутренние ЗУ. При формировании терминов и определений использованы ГОСТ 15971-74 и ОСТ 11 305.902-80.

Микропроцессорная электронная вычислительная машина (микроЭВМ) — цифровая электронная вычислительная машина с ин-

---

* Малашевич Б. М., Шахнов В. А., Коночкин Э. И. Термины и определения: Микропроцессорные средства и системы. Микропроцессорные интегральные микросхемы. — «Микропроцессорные средства и системы», 1984, № 3, с. 90—92.



Ветвь «Микропроцессорные модули» терминологического дерева:
«Микропроцессорные средства и системы»

терфейсом ввода-вывода, состоящая из микропроцессора, полупроводниковой памяти и, при необходимости, пульта управления и источников электропитания, объединенных общей несущей конструкцией.

В последние годы получает распространение синоним термина микроЭВМ — микрокомпьютер (microcomputer). Часто термины «микроЭВМ» и «ЭВМ» некорректно распространяют на микропроцессорные системы, например, в таких относительно устоявшихся словосочетаниях, как «персональная ЭВМ». Такое вольное отношение нарушает строгость терминологии, приводит к путанице и нарушению классификации изделий.

Семейство микроЭВМ — совокупность архитектурно совместимых микроЭВМ. Под архитектурной совместимостью понимается совместимость *программного обеспечения* и *интерфейсов*.

Программы, написанные для одной микроЭВМ, должны полностью и без доработки выполняться на другой ЭВМ семейства, если последняя по своим функциональным возможностям равна или превосходит последнюю (совместимость снизу вверх), т. е. программы младших моделей семейства должны выполняться на равных или старших моделях семейства. В рамках семейства должна быть реализована единая система интерфейсов, обеспечивающая возможность использования при построении микропроцессорных систем единого парка микропроцессорных модулей и других устройств.

Встраиваемая микроЭВМ — микроЭВМ, не имеющая индивидуального пульта управления, источника электропитания, декоративного оформления и предназначенная для конструктивного встраивания в вычислительную или управляющую микропроцессорную систему.

Одноплатная микроЭВМ — встраиваемая микроЭВМ, выполненная в виде микропроцессорного модуля, построенного на основе одной печатной платы.

Автономная микроЭВМ — микроЭВМ, имеющая законченное конструктивно-декоративное оформление, встроенный источник электропитания и, при необходимости, простейший пульт управления.

Микропроцессор — программно-управляемое устройство, осуществляющее процесс **обработки** цифровой информации и **управления** им, построенное на одной или нескольких интегральных микросхемах.

В состав микропроцессора входят ***операционный блок,*** выполняющий арифметические и логические операции, ***блок микропрограммного управления,*** определяющий алгоритм функционирования операционного блока в соответствии с кодом поступившей команды, и ***интерфейсный блок,*** обеспечивающий

сопряжение микропроцессора с системной магистралью.

Центральный микропроцессор — микропроцессор, непосредственно осуществляющий процесс обработки данных в однопроцессорной системе или организацию работы неоднородной мультипроцессорной системы.

Периферийный микропроцессор (сопроцессор) — микропроцессор, выполняющий под управлением центрального микропроцессора определенную функцию в микропроцессорной системе.

Периферийный микропроцессор может быть либо идентичен центральному, либо специализирован.

Специализированный микропроцессор — микропроцессор, структура которого оптимизирована для решения определенного класса задач.

Арифметический микропроцессор — специализированный микропроцессор, структура которого оптимизирована для выполнения арифметических операций.

В последние годы в печати стал встречаться вводящий в заблуждение термин «аналоговый процессор». На самом деле имеется в виду обычный цифровой микропроцессор или микроЭВМ с встроенными в кристалл микросхемы преобразователями для ввода (АЦП) и вывода (ЦАП) аналоговой информации. Поэтому правильнее такие изделия относить к однокристальным микропроцессорным системам.

Разрядность микропроцессора — число разрядов регистров арифметического логического устройства (АЛУ) микропроцессора.

Разрядность микропроцессора не всегда соответствует разрядности обрабатываемых данных. Например, 8-разрядное АЛУ микропроцессорного комплекта БИС К581 осуществляет (за два такта) обработку 16-разрядных данных.

Память электронной вычислительной машины — функциональная часть ЭВМ, предназначенная для запоминания и (или) выдачи данных.

В зависимости от функционального назначения память может быть оперативная, буферная и т. д.

Запоминающее устройство (ЗУ) — изделие, реализующее память.

В зависимости от используемой запоминающей среды может быть полупроводниковое, ферритовое, пленочное и т. д.

Внутреннее запоминающее устройство — запоминающее устройство, память которого расположена внутри адресного пространства микропроцессора.

Адресное пространство микропроцессора — совокупность адресов внутренних регистров, внутренних ЗУ и регистров периферийных устройств, для обращения к которым достаточно содержимого регистра адреса микропроцессора.

Оперативное запоминающее устройство (ОЗУ) — внутреннее запоминающее устройство, обеспечивающее возможность оперативного изменения информации, используемое для записи, хранения и выдачи информации, в том числе во время выполнения программы, и имеющее длительность цикла обращения, соизмеримую с длительностью цикла выполнения микропроцессором основных операций.

Постоянное запоминающее устройство (ПЗУ) — запоминающее устройство с неизменяемым содержимым памяти.

В полупроводниковые ПЗУ информация записывается в процессе изготовления микросхемы путем соответствующего соединения запоминающих элементов на поверхности кристалла.

Программируемое постоянное запоминающее устройство (ППЗУ) — постоянное запоминающее устройство, *в которое информация заносится однократно потребителем* не в составе изделия и не может быть впоследствии изменена.

В полупроводниковое ППЗУ информация заносится путем пережигания плавких перемычек на поверхности кристалла. Ранее под ППЗУ понималось полупостоянное запоминающее устройство. В настоящее время этот термин практически не применяется, вместо него употребляется термин *«репрограммируемое ПЗУ»* или *«перепрограммируемое ПЗУ»*. Первый из них предпочтительнее.

Репрограммируемое постоянное запоминающее устройство (РПЗУ) — постоянное запоминающее устройство, *в котором информация может неоднократно изменяться* при помощи специальных средств стирания и записи.

Для записи новой информации микросхема РПЗУ, как правило, извлекается из изделия и соединяется со специальным программатором. Стирание информации осуществляется при помощи ультрафиолетового облучения (УФ РПЗУ), электрически (ЭРПЗУ) и т. п.

Энергонезависимое оперативное запоминающее устройство — оперативное запоминающее устройство, в котором информация сохраняется при отключении электропитания.

Энергонезависимость обеспечивается либо особенностями запоминающего элемента, либо введением аварийного электропитания на основе аккумуляторов, ионисторов и т. п. При этом энергопотребление ОЗУ в режиме хранения, как правило, значительно ниже, чем в режиме обращения.

Статья поступила 17 октября 1984 г.

РЕФЕРАТЫ СТАТЕЙ

УДК 681.32

Малашевич Б. М., Дшхунян В. Л., Борщенко Ю. И., Никольский О. А., Хорин В. С. **Магистрально-модульные микропроцессорные системы.** — Микропроцессорные средства и системы, 1984, № 4, с. 3—11.

Рассмотрены особенности реализации магистрально-модульного принципа построения радиоэлектронной аппаратуры на основе микропроцессоров. Описывается иерархия модулей, требования к ним и обобщенные характеристики наиболее перспективных однокристальных и вдвижных модулей.

UDC 681.32

Malashevich B. M., Dshkunyan V. L., Bortshenko Yu. I., Nikolsky O. A., Horin V. S. Modular Microprocessor Systems. — Microprocessor Devices and Systems, 1984, N 4, p. 3—11.

The paper discusses the principles of microprocessor architecture based on the notion of system bus with modules. The hierarchy of modules is discussed and their general characteristics are considered in view of the main trends of microprocessor systems.

УДК 681.323—181.48

Дшхунян В. Л., Борщенко Ю. И., Науменков В. Р., Рыжов А. А., Романец Ю. В., Бурмистров И. А., Соловьев Е. М. **Однокристальные микропроцессоры и запоминающие устройства комплекта БИС серии К1801.** — Микропроцессорные средства и системы, 1984, № 4, с. 12—18.

Приведены основные электрические характеристики и параметры функционирования однокристальных микропроцессоров перспективного микропроцессорного комплекта БИС серии К1801. Подробно описаны системный интерфейс и назначения выводов микросхем. Представлены структурные схемы и временные диаграммы режимов обмена информацией.

UDC 681.323—181.48

Dshkunyan V. L., Bortshenko Yu. I., Naumenkov V. R., Ryzhov A. A., Romanetz Yu. V., Burmistov I. A., Solovjov E. M. K1801 Family of Singlechip Microprocessors and Memory Circuits. — Microprocessor Devices and Systems, 1984, N 4, p. 12—18.

The main electrical and functional characteristics of the LSI family K1801 are given in the paper. The detailed discussion of microprocessor and memory system interfaces, IC's pin description, logical and timing diagrams are presented.

УДК 681.3—181.4

Алексеев Ю. Д., Бойченко А. В. **Периферийные устройства мини- и микроЭВМ.** — Микропроцессорные средства и системы, 1984, № 4, с. 19—25.

В обзоре на основе существующей классификации технических средств мини- и микроЭВМ рассматриваются наиболее характерные функциональные группы периферийных устройств — внешние запоминающие устройства на магнитных носителях и устройства ввода-вывода информации, в частности печатающие устройства различных типов. Анализируется современное состояние и основные технические характеристики этих устройств.

Обзор предназначен для широкого круга специалистов, занимающихся применением мини- и микроЭВМ и управляющих вычислительных комплексов на их основе в системах управления различного назначения.

UDC 681.3.—181.4

Alekseev Yu. D., Boytchenko A. V. Peripheral Devices of mini- and microcomputers. — Microprocessor Devices and Systems, 1984, N 4, p. 19—25.

The paper considers the typical groups of peripheral devices for mini — and microcomputers — external storage devices, printers etc.

The general technical specifications of each group of devices are given. The survey is intended for the specialists involved in mini- and microcomputer application in control systems.

УДК 681.327.8(3,82,7)+681.324—181.4

Виноградов Б. Н., Шахнов В. А. **Распределенные микропроцессорные системы и локальные вычислительные сети ЭВМ.** — Микропроцессорные средства и системы, 1984, № 4, с. 26—31.

Рассмотрены сетевые принципы организации распределенных вычислительных систем с использованием техники локальных вычислительных сетей (ЛВС). Указаны структурные и архитектурные особенности ЛВС. Приведены основные типы и характеристики систем передачи данных, даны особенности их применения в различных распределенных ВС.

UDC 681.327.8(3,82,7)+681.324—181.4

Vinogradov V. N., Shakhnov V. A. Distributed Microprocessor Systems and Local Area Networks. — Microprocessor Devices and Systems, 1984, N 4, p. 26—31.

The authors consider the principles of distributed computer systems based on Local Area Networks (LAN) approach. The structure and architectural features of different LANs are considered. The basic types and specification of communication systems and their application in distributed computer systems are discussed.

УДК 681.327.8:537.533.3

Ананян М. А., Мельникова О. В. **Оптоволоконные локальные сети.** — Микропроцессорные средства и системы, 1984, № 4, с. 32—35.

В краткой форме рассмотрены системы информационного обмена и возможности их реализации с помощью оптоволоконных локальных сетей, построенных на основе выпускаемых отечественной промышленностью систем «Электроника МС 4101», «Электроника МС 8201», «Электроника МС 8401», «Электроника МС 8050».

UDC 681.327.8:537.533.3

Ananjan M. A., Melnikova O. V. Fiber-Optic Local Area Networks. — Microprocessor Devices and Systems, 1984, N 4, p. 32—35.

An overview of fiber-optic based local area networks is presented. The industrial systems «Electronica MC 4101, 8201, 8401 and 8050» are considered as the technical basis for implementation of such networks.

УДК 681.327.8

Долкарт В. М., Карпенко С. Л., Курцман Г. М., Перцов Е. Е., Редина С. Ф. **Дистанционная магистраль — локальная управляющая вычислительная сеть микросистем В7/В9.** — Микропроцессорные средства и системы, 1984, № 4, с. 36—39.

Рассмотрена архитектура дистанционной магистрали локальной управляющей вычислительной сети В7/В9, предназначенной для сложных управляющих систем с распределенной мощностью.

UDC 681.327.8

Dolkart V. M., Carpenko S. L., Curzman G. M., Pertzov E. E., Redina S. F. Extended Bus — the tocal Area Control Network for V7/V9 microsystems. — Microprocessor Devices and Systems, N 4, p. 36—39.

The architecture of extended (distant) bus is considered which is used for building V7/V9 microsystems-based LAN. It is intended for distibuted control of complex systems.

УДК 681.326.3:681.327.8

Мячев А. А., Снегирев А. А. **Контроллеры приборных интерфейсов мини- и микроЭВМ.** — Микропроцессорные средства и системы, 1984, № 4, с. 39—43.

Рассмотрены общие вопросы создания семейства взаимосвязанных средств сопряжения системных магистралей типа общая шина с магистралями программно-модульных приборных интерфейсных систем КАМАК и МЭК 625-1. Проведен анализ средств сопряжения магистралей, выбраны основные показатели назначения контроллеров, дана кассификация и рассмотрена структура организации контроллеров различных типов, предложены критерии и методы оценки эффективности контроллеров. Дается общее описание структуры контроллеров в стандарте КАМАК, реализованных на евроконструктивах одноплатных контроллеров МЭК 625-1, универсальных каналов прямого доступа к памяти и периферийным устройствам, выходящим на общую шину мини- и микроЭВМ.

UDC 681.326.3:681.327.8

Myachev A. A., Snegirev A. A. Instrumentation interface Controllers for Mini- and Microcomputers. — Microprocessor Devices and Systems, 1984, N 4, p. 39—43.

The author discusses problems of interface development for communication between computer system buses and instrumentation systems based on CAMAC and IEC 625-1 standards. The analysis of different technical approaches is presented, specification and logical structure of several controllers are given, criteria for their selection are proposed. The general description is given for CAMAC and IEC 625-1 controllers, universal direct memory access controllers and other devices.

УДК 681.3.06

Игнатьев М. Б., Кибиткин В. В., Осовецкий Л. Г. **Структура и особенности системного программного обеспечения микроЭВМ.** — Микропроцессорные средства и системы, 1984, № 4, с. 44—47.

Дается комплексное описание системного программного обеспечения микроЭВМ на уровне функциональной организации. Рассматриваются особенности и взаимосвязь операционных систем программирования и инструментальных средств программирования для микроЭВМ. Формулируются основные задачи по развитию работ в рассматриваемой области.

UDC 681.3.06

Ignatjev M. B., Kibitkin V. V., Osovetzky L. G. The Structure and Characteristics of Software for Microcomputers. — Microprocessor Devices and Systems, 1984, N 4, p. 44—47.

Microcomputer software is described from functional standpoint. The characteristics of operating systems and instrumental programming systems are given, their interfaces are discussed. The main problems of software development are being considered.

УДК 681.3.06

Борковский А. Б. **Многооконное текстовое взаимодействие с персональной ЭВМ.** — Микропроцессорные средства и системы, 1984, № 4, с. 47—50.

Дано внешнее описание и обоснование подхода к организации взаимодействия с персональной ЭВМ с помощью системы управленяи окнами на экране, каждое из которых связано с какой-либо работой; изложены принципы реализации и конкретная реализация такой системы.

UDC 681.3.06

Borkovsky A. B. Multiwindow Textual Interaction with Personal Computers. — Microprocessor Devices and Systems, 1984, N 4, p. p. 47—50.

The author describes the method of communicating with a personal computer througt te multi-window display system. Each window becomes associated with particular job. The general principles and implementation of multi-window system is described in the article.

УДК 681.3.06

Хоар Ч. А. Р. **Программирование как инженерная профессия.** — Микропроцессорные средства и системы, 1984, № 4, с. 53—60.

Дан сравнительный анализ инженерных наук и программирования на стадиях разработки, реализации проекта и сдачи в эксплуатацию программы. Рассмотрены вопросы выбора надежных методов и «инструментов» для программирования.

UDC 681.3.06

Hoar C. A. R. Programming as engineering Science. — Microprocessor Devices and Systems, 1984, N 4, p. 53—60.

A translation of C. A. R. Hoar's paper, where the author presents a comparative study of engineering sciences and programming with respect to the stages of project development, implementation and practical accomplishment. The problem of choosing reliable methods and «tools» is also considered.

УДК 681.322.1

Попов А. А., Хохлов М. М., Глухман В. Л. **Диалоговые вычислительные комплексы «Электроника НЦ-80 — 20».** Микропроцессорные средства и системы, 1984, № 4, с. 61—64.

Рассматриваются основные принципы разработки ДВК, вопросы архитектуры и программного обеспечения. Дается краткая характеристика моделей, приводится состав и структурные схемы ДВК.

UDC 681.322.1

Popov A. A., Khohlov M. M., Gluhman V. L. «Electronica NC-80—20»: the dialogue Computer System. — Microprocessor Devices and Systems, 1984, N 4, p. 61—64.

The paper describes the main principles, archtectural and software features of the DVK («Electronica NC-80—20») microcomputer. A survey of the basic models and their technical specification is presented.

УДК 681.32—181.4:621.31

Колобродов Н. В., Кусакина Л. С., Мельников В. С., Федулова О. А., Чугунников Ю. В. **Микропроцессорное устройство управления мощностью крупных паровых турбин.** — Микропроцессорные средства и системы, 1984, № 4, с. 65—68.

Рассматриваются аппаратные и программные средства микропроцессорной системы, реализующей функции всережимного регулятора мощности крупных паровых турбин.

UDC 681.32—181.4:621.31

Kolobrodov N. V., Kusakina L. S., Fedulova O. A., Tchugunnikov Yu. V. Microcomputer Controller in the Power Refulation System of Large Steam Turbines. — Microprocessor Devices and Systems, 1984, N 4, p. 65—68.

Microprocessor system hardware and software is considered which performs the full-range regulation of steam turbines power output.

УДК 658.012

Кочетков А. А., Крылов В. В. **Диалоговая система для автоматизации метеорологических измерений.** — Микропроцессорные средства и системы, 1984, № 4, с. 68—72.

Рассматривается архитектура аппаратных и программных средств проблемно-ориентированной диалоговой системы дистанционного измерения высотных профилей метеорологических параметров атмосферы радиометрическим методом. Предлагается схема организации диалога с циклическим графом, которая позволяет уменьшить число ответов пользователя при частичной настройке многопараметрической системы, по сравнению с обычным древовидным графом. Описывается простой диспетчер оперативной и фоновой задач, позволяющий обеим задачам использовать общую библиотеку подпрограмм с целью экономии памяти Система построена на базе микроЭВМ «Электроника 60». Программное обеспечение реализовано на языках Фортран и ассемблере.

Kochetkov A. A., Krylov B. B. The Dialogue System for Automation of Meteorological Reseach. — Microprocessor Devices and Systems, 1984, N 4, p. 68—72.

The hardware and software components of the dialogue system are described intended for automation of telemeasuring equipment used in atmospere research. The dialogue scheme have a form of a cyclic graph as opposed to the more traditional tree structured graph. The system is based on «Electronica 60» microcomputer with the software, including a symple monitor and a library of subprograms in Fortran and Assembler.

УДК 681.32—181.4:621.31

Ковалев В. Д., Хансуваров А. К., Шевченко А. Т. **Микропроцессорная система противоаварийного управления повышенной надежности.** — Микропроцессорные средства и системы, 1984, № 4, с. 73—76.

Рассматриваются программные и аппаратные средства микропроцессорной системы для обеспечения устойчивости параллельной работы электростанций. Подробно изложены средства повышения надежности функционирования.

UDC 681.32—181.4:621.31

Kovalyov V. D., Khansuvarov A. K., Shevtcenko A. T. Microprocessor — Based Antiwreck Control System of High Reliability. — Microprocessor Devices and Systems, 1984, N 4, p. 73—76.

Microprocessor — based system is considered intenden for electric power station control. The stability and reliability of their parallel functioning is a matter of prime importance.

Указатель статей, помещенных в журнале „Микропроцессорные средства и системы" в 1984 г.

Микропроцессорная техника

Персональные компьютеры

Программное обеспечение

Применение микропроцессорных средств

Учебный центр

В странах — членах СЭВ

Термины и определения

Выставки

Пятиугольник «МП»

Редакционную почту с откликами на статью Д. О. Цента «Электронный ключ в функционально полной системе ключей» («МП» № 3, 1984, с. 96) можно, видимо, условно разделить на четыре заметно разные по высоте стопки писем. Первой и самой большой (87% общего числа писем) оказалась стопка с конвертами и открытками, в которых читатели выражают признательность за дополнительные и более четкие разъяснения по вопросам стиля, характера изложения и, главное, содержания материалов, которые следует (или НЕ следует) направлять в «МП» для публикации. Таким образом, подавляющая часть наших читателей подошла к тексту 96-й полосы «МП» № 3 с конструктивных позиций и восприняла статью Д. О. Цента как иллюстрацию к 96-й полосе «МП» № 2, содержащую «Памятку автору».

Вторая стопка писем (6% их общего числа) представляет собой резкие или наоборот подчеркнуто вежливые предупреждения об опасностях, которые ожидают редакцию молодого журнала в случае, если будет продолжаться разглашение на его 96-й полосе профессиональных тайн, обеспечивающих, судя по письмам, их авторов, длительное и до сих пор никем и ничем не омрачаемое процветание на ниве «греко-латинской» формализации околокомпьютерных наук.

Третью группу писем (4%) составляют конкретные критические замечания, адресованные непосредственно автору опубликованной статьи — Д. О. Центу. Читатели, а это оказались видимо наиболее внимательные из наших подписчиков, справедливо указывают на ряд досадных неточностей в формальных выкладках автора, а также перечисляют фактические ошибки в использовании им методики подсчета экономической эффективности выполненной на кафедре работы. Как удалось установить нашим читателям, действительная сумма экономического эффекта была значительно завышена Д. О. Центом — по некоторым оценкам почти на 17% по сравнению с теми точными данными, которые следуют из последней версии, рекомендованной кафедрам методики расчета экономической эффективности (см. например, ХНО/013—85). Все эти конкретные критические письма после обработки, классификации и машинной каталогизации будут направлены (по спутниковым каналам электронной почты) для принятия мер непосредственно по ведомственной принадлежности автора. О времени пуска в регулярную эксплуатацию редакционного канала спутниковой электронной почты журнал сообщит подписчикам дополнительно*.

И, наконец, в четвертой и самой небольшой по объему (около 3%) стопке писем содержится подробный анализ литературных источников статьи Д. О. Цента. Выяснилось, в частности, что Д. О. Цент, кроме вскользь упомянутых им во введении трудов конференции, существенно использовал в своих, как выше мы уже отмечали, далеко не безупречных, математических выкладках и текстовом описании незаслуженно малоизвестную статью «Ключ в системе ключей», опубликованную 20 лет назад в одном из ставших уже библиографической редкостью физических сборников (Обнинск, май 1965, с. 205—208).

Именно эти читатели и оказались победителями 1-го тура заочного конкурса: **«Знаете ли Вы литературные истоки фундаментальных основ техники процветания на ниве псевдоматематизации околокомпьютерных наук?»**

По итогам работы жюри конкурса принято следующее решение:

1. Имена всех победителей — 172 человека из 53 городов — занесены в электрически прожигаемое ПЗУ электронного ключа, укрепленного на входной двери редакции «МП» (пр. Серова, д. 5, 1-е крыльцо налево).
2. Редакционный профессиональный персональный компьютер «Тольконеагат-1» 1.04.85 начинает генерацию 3-х машинных поэм и 169 звукосинтезируемых кантат, которые будут затем высылаться адресатам на пятидюймовых дискеттах. Поэтому всем участникам конкурса, претендующим на призовые места, следует заблаговременно направлять дискетты типа ЕС 5288, BASF и т. п. в адрес редакции с почтовой пометкой: «Вручить 1.04.85».

Условия 2-го тура конкурса будут объявлены в следующем номере.

По поручению жюри конкурса

Г. Р. Громов

---

* Запуск на геостационарную орбиту очередного спутника связи «МП-1001» до сих пор, к сожалению, неоднократно откладывался, в том числе по метеоусловиям, сложившимся за первый год работы в помещении редакции «МП». Недопустимо высокий уровень задымленности (суммарный уровень прозрачности редакционной атмосферы не превышает отметки 32 по шкале Кэмела, т. е. определяется темпом сгорания 32 сигарет за каждые четверть часа общения с авторами). Кроме того, некоторые сложности создает в этом отношении достигнутая журналом рекордно высокая плотность размещения персонала: 6 человек штатных сотрудников редакции, шкафы, ЭВМ, рецензентов, авторов с макетами описываемых устройств и любознательных читателей, оказалось возможным разместить на 15 квадратных метрах, включая 3 ступеньки входного крыльца, что также до некоторой степени ограничивает возможность оперативного развертывания стартового комплекса, даже в обеденный перерыв или ночное время.

МИ

ИЗД



Антон
главн
«Маш

МИКРОПРОЦЕССОРНАЯ ТЕМАТИКА В ПЛАНАХ ИЗДАТЕЛЬСТВА «МАШИНОСТРОЕНИЕ»



Антонов Борис Игоревич, главный редактор издательства «Машиностроение»

Каждый, кто посетил проведенную в Москве выставку «Металлообработка-84» — будь то специалист машиностроения или смежной отрасли — не мог не обратить внимания на широкое использование технических средств и решений, бывших еще недавно достоянием фундаментальных наук. Это лазерная техника, используемая как средство обработки машиностроительных материалов и высокоточных измерений, волоконная оптика и многое другое. Но, бесспорно, что определяющее влияние на современный уровень машиностроения оказывают достижения электроники, в частности микропроцессорной техники. Именно машиностроение и особенно его сердцевина — станкостроение — обеспечивают этой технике широкомасштабное промышленное внедрение.

Естественно, это нашло отражение в выпускаемой издательством литературе, и прежде всего, в работах по станкам с числовым программным управлением (ЧПУ). Так, в 1983 г. были выпущены две книги: «Основы программирования обработки на станках с ЧПУ» (автора Г. В. Евгеньева) и «Микроэлектронные устройства в системах управления станками» (коллектива авторов под редакцией С. Г. Синичкина). В последней работе, входящей в серию «Основы электроавтоматики станков», рассказано о возможностях управления станками с помощью различных микроэлектронных устройств — аналоговых и цифровых интегральных микросхем и, особенно БИС.

Совместное издание советских и румынских специалистов (Л. Г. Филиппова, И. Р. Фрейдзон, А. Давидовичу, Э. Дятку) «Мини- и микроЭВМ в управлении промышленными объектами» выпущено ленинградским отделением нашего издательства и бухарестским издательством «Техника» в первом полугодии 1984 г. В этой работе дана характеристика отечественных и зарубежных микропроцессоров различных поколений, показаны принципы управления и организации микроЭВМ, создаваемых на их основе, программное обеспечение, проектирование систем автоматизации, применение их для автоматизации промышленных объектов и процессов. Микропроцессорам уделяется должное место в наших изданиях по роботам, робототехническим системам, гибким автоматическим производствам.

В ближайшее время выйдет в свет книга В. Л. Сосонкина «Микропроцессорные системы числового программного управления станками». При создании подобной техники все чаще требуются специалисты широкого профиля, способные ориентироваться в сложных вопросах на стыке различных областей знания. Например, математическое обеспечение микропроцессорных устройств ЧПУ. Действительно, способны ли «чистые» математики и системные программисты решить весь комплекс вопросов, относящихся к этой проблеме, носящей ярко выраженную технологическую окраску? Способны ли технологи, специалисты в области автоматизации станков решать задачи, требующие профессионализма в системном программировании? Мы надеемся, что книга В. Л. Сосонкина в какой-то мере послужит решению возникшей проблемы.

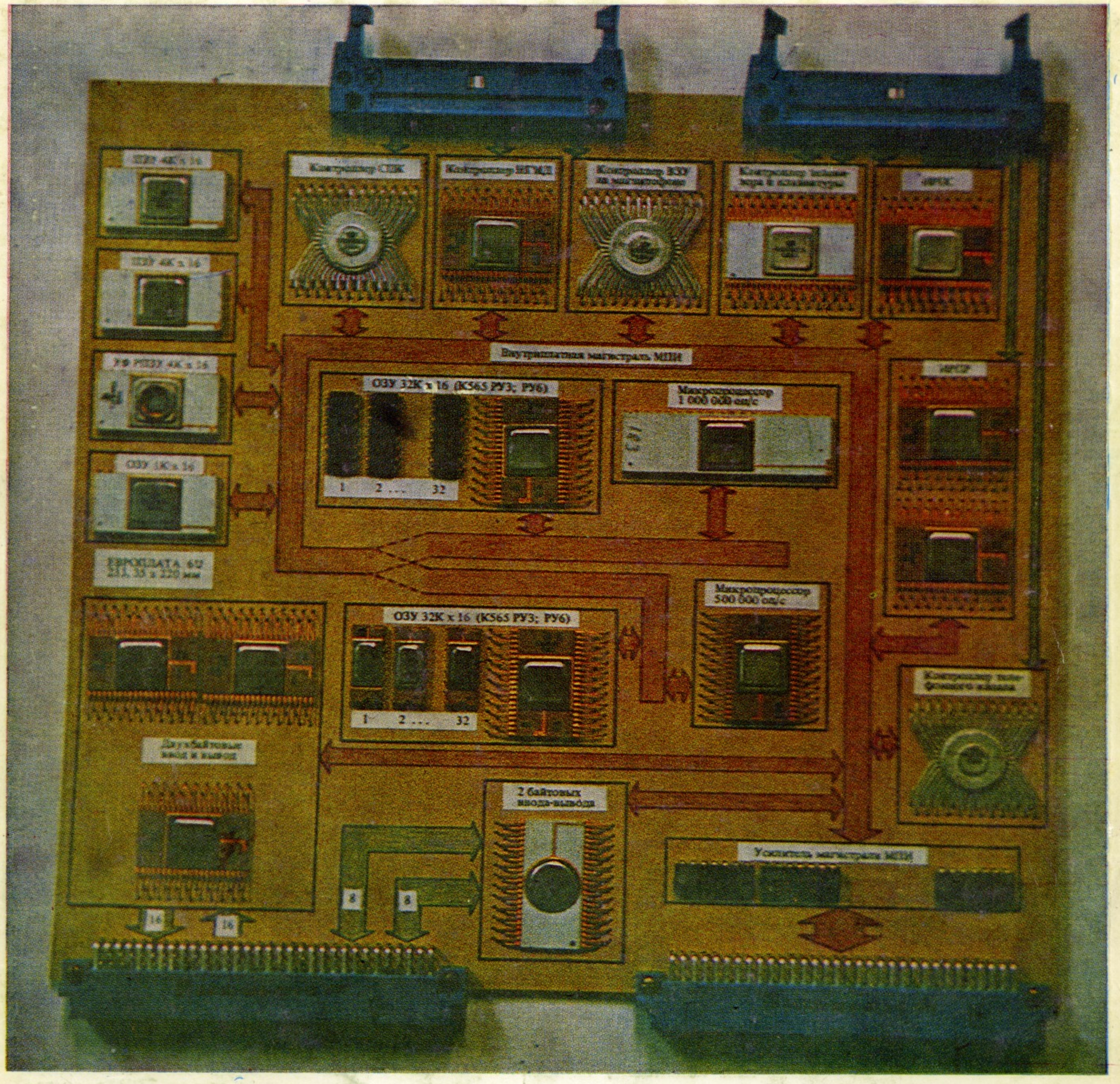
В этой работе описаны функциональные особенности микропроцессорных устройств ЧПУ, их структурные и схемотехнические особенности, базовые математические средства. Большое внимание уделено вопросам математического обеспечения и, наконец, интеграции металлорежущих станков с микропроцессорным управлением в гибких производственных системах.

Вопрос о применении микропроцессорных средств в ГАП рассмотрен в книге В. А. Ратмирова «Управление станками гибких производственных систем».

Следует также отметить работу «Автоматизация проектирования аналого-цифровых приборов на микропроцессорах» (автора Б. Л. Собкина) — первую книгу по автоматизации проектирования приборов и устройств на базе микропроцессорных комплектов БИС. Выпуск этой книги намечен на 1986 год.

Издательство «Машиностроение» из года в год будет увеличивать выпуск книг, непосредственно связанных с микропроцессорами. Наша ближайшая задача — дать специалистам машиностроения и приборостроения пособие, которое поможет им лучше ориентироваться в микропроцессорной технике. В настоящее время коллектив ленинградских авторов работает над справочником «Микропроцессоры и ЭВМ в системах автоматического управления». В нем будут систематизированы основные сведения по микропроцессорам и комплектам БИС, изложены основы организации микропроцессоров, микроЭВМ и систем на их основе, а также другие вопросы, необходимые для специалистов, занимающихся созданием систем ЧПУ, САПР, ГАП, АСУ ТП, автоматизированных систем технологической подготовки производства и другие. Мы надеемся, что эта работа будет с интересом встречена и машиностроителями, и специалистами смежных отраслей.

1р. 10к. Индекс 70588



ЧИТАЙТЕ В СЛЕДУЮЩЕМ НОМЕРЕ

- Скоростной дуплексный адаптер «Электроника МС 8002» для создания многомашинных вычислительных комплексов на базе мини- и микроЭВМ, имеющих единый канал типа «Общая шина»: «Электроника 100/25», «Электроника 79», СМ-4 и др.
- Секционированный 8-разрядный микропроцессорный комплект серии КР583, выполненный на основе интегральной инжекционной логики
- Как сделать и отладить различные варианты последовательного интерфейса на базе БИС КР580ВВ51?
- Модули на базе микропроцессорных БИС КР580, используемые для автоматизации исследований в области физики высоких энергий
- Учебно-отладочная микроЭВМ «Электроника ОУ—580» — инструмент для практического обучения работе с наиболее массовым микропроцессорным комплектом БИС КР580